



# D30 EVO DASH

## MANUAL



2014 GET by Athena

The contents of this document, or any part of it, may not be reproduced, transferred, distributed or memorised in any form whatsoever without the written permission of GET by Athena.

GET reserves the right to modify the content of this manual without prior notice.

## TABLE OF CONTENTS:

1	ADVICES AND CAUTIONS DURING INSTALLATION .....	5
2	D30EVO: MECHANICAL SYSTEMS .....	6
3	D30EVO: COMMANDS AND INDICATORS .....	7
4	D30EVO: PINOUT CONNECTORS .....	8
5	D30EVO: SYSTEM CONNECTION .....	9
5.1	Wiring code GL-0092-AA .....	9
5.2	Wiring code GL-0086-AA .....	10
6	TECHNICAL CHARACTERISTICS .....	11
7	USE OF THE D30EVO DASHBOARD .....	12
7.1	Functional keys .....	12
8	PRINCIPLES FOR THE CONFIGURATION OF THE D30EVO DASHBOARD .....	13
8.1	BAR type field .....	15
8.2	LABEL type field .....	16
8.3	VALUE type field .....	17
8.4	Signals (ALARM) .....	18
9	CONFIGURATION OF D30EVO: PRELIMINARY ACTIONS .....	19
9.1	Suggestions for the configuration of MAYA .....	19
9.2	Connection of the dashboard to the PC .....	20
9.3	Opening a project (configuration) file from the PC .....	22
9.4	Downloading the configuration of the dashboard .....	23
9.5	Saving a project .....	25
9.6	Programming the D30EVO: .....	26
10	CONFIGURATION OF D30EVO .....	29
10.1	Dash Channel Map .....	31
10.2	Page1...Page5 Map .....	31
11	HOW TO... .....	32
11.1	Changing the scale of the motor routes .....	32
11.2	Changing the change flash .....	33
11.3	Changing the alarm thresholds .....	34
11.4	Setting a CAN channel in reception .....	35
12	DEVICE TABLES .....	36



Dear customer,

Thank you for having chosen a product of the DATA ACQUISITION AND ANALYSIS SYSTEMS line of **GET** by **Athena**.

We are confident that our passion and experience may help you successfully express yourself in any competition you might enter and we invite you to read the hereby manual, which will surely help you use correctly your new device **GET** by **Athena**.

Developed for the requirements of the motor sports sector, **D30 EVO** is distinguished through its high robustness, its accurate design, its easy installation and maximum usage flexibility.

The extended display clearly views data for the pilot and the signalling mechanisms warn him/her in real time on the presence of any alerts. Therefore, the **D30EVO** dash is an irreplaceable support in borderline situations.

Due to the integrated CAN bus interface, the installation is quick and simple: only four electrical connections are required for rendering the system operational at a hardware level.

The configuration of the dashboard is assigned to the dedicated management software that allows for the customization of viewed data and alerts.

Self-adhesive labels (included in the kit) help mark the alert leds.

The hereby manual aims at providing the use with a guide on the functions of D30EVO dashboards: please read carefully the hereby document for a correct use of the product.

## 1 ADVICES AND CAUTIONS DURING INSTALLATION

Before installing **D30EVO** dashboards on a vehicle, please follow the guidelines below:

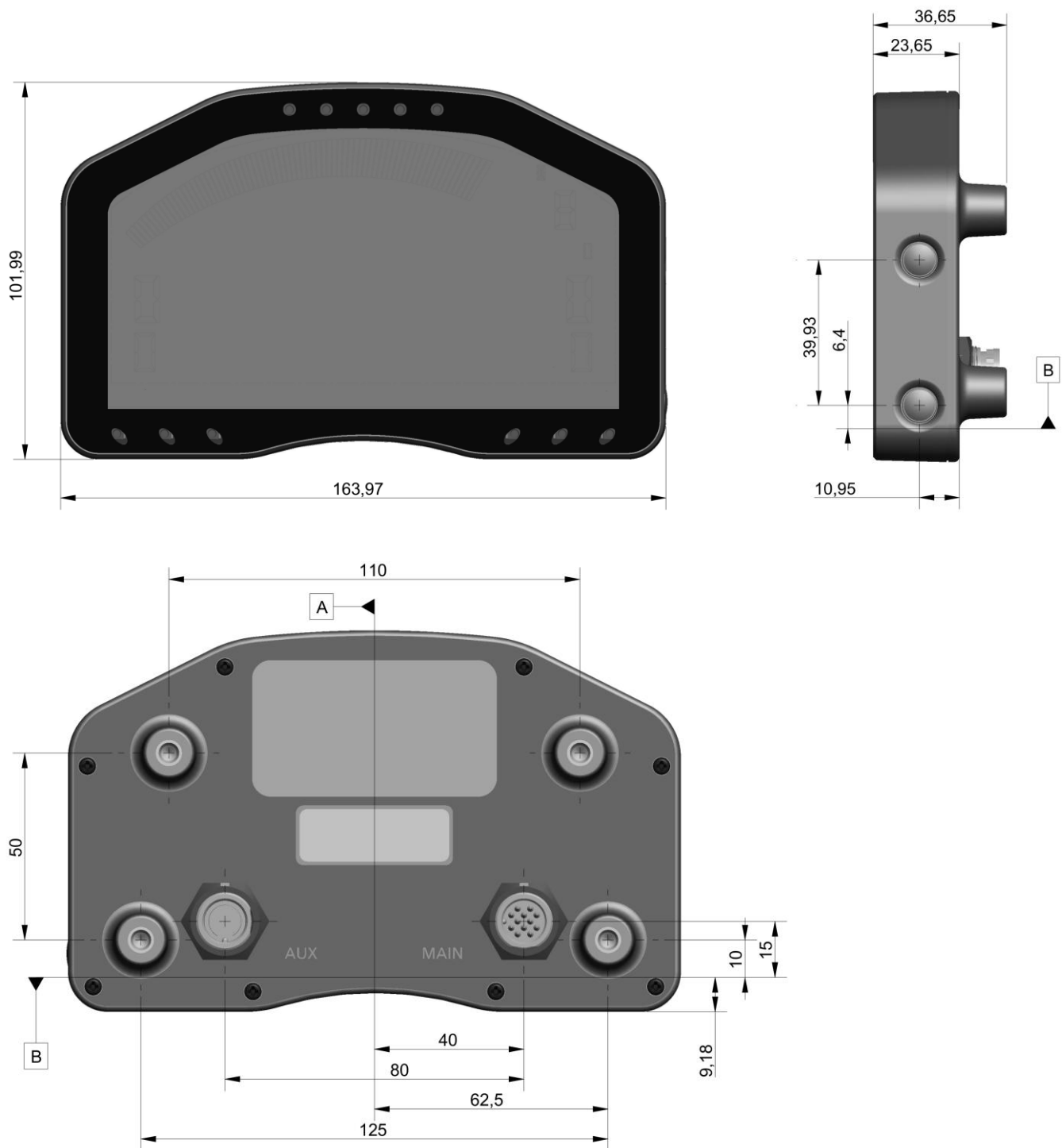
- Perform assembly/disassembly operations in a place with adequate space.
- Then disconnect the poles of the vehicle's battery.
- Any parts of the vehicle that may have been disassembled should be kept in a place where there is no risk that they might be damaged.
- The installation of the system in vehicles with endo-thermal engine must be performed on an idle engine: during assembly, contact may occur with the parts of the engine or framework that are subject to heating
- Make sure not to damage connectors and wirings during assembly/disassembly.
- Make sure you don't lose screws and washers inside the vehicle during installation
- During installation, make sure that no installed part interferes with the guidance devices of the vehicle or with the pilot



**CAUTION: A FAULTY INSTALLATION MAY RESULT IN SERIOUS DAMAGE TO INDIVIDUALS AND/OR OBJECTS**

## 2 D30EVO: MECHANICAL SYSTEMS

In the following the mechanical systems of the **D30EVO** dashboard:

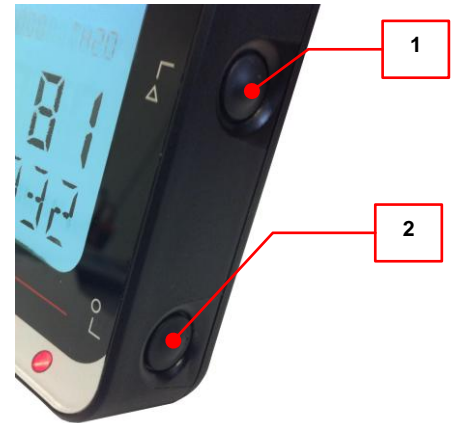


QUOTE IN mm



### 3 D30EVO: COMMANDS AND INDICATORS

- 1: Button  $\Delta$
- 2: Button  $\circ$
- 3: Shift Light LEDs
- 4: Display
- 5: Alarm LEDs
- 6: Alarm LEDs
- 7: Connector **AUX**
- 8: Connector **MAIN**

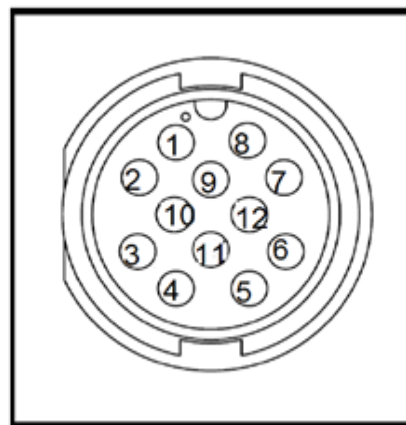


## 4 D30EVO: PINOUT CONNECTORS

In the following the description of the connectors of the **D30EVO** dashboard:

### MAIN CONNECTOR:

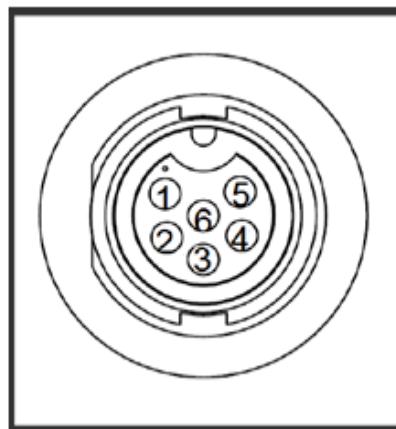
PIN	NAME	DESCRIPTION
1	VBB1	Positive supply (VBB1)
2	GND POW	Negative supply
3	CANL	CAN bus – CANL signal
4	CANH	CAN bus – CANH signal
5	CANT	CAN bus - termination
6	GND SEN	Analogic input ground
7	RS_TX1	RS232 serial port – D30 out
8	RS_RX1	RS232 serial port – D30 in
9	PRG	Programming signal input
10	AN1	AN1 analogic input
11	DIN1	DIN1 frequency input
12	BEACON	Beacon input



NOTE: connect the CANT signal to CANH if you want to use the internal termination resistance of the **D30EVO** dashboard (120Ω).

### AUX CONNECTOR:

PIN	NAME	DESCRIPTION
1	VBB AUX	Supply output (similar to VBB1)
2	GND SEN	Analogic input ground
3	CANL	CAN bus – CANL signal
4	CANH	CAN bus – CANH signal
5	AN2	AN2 analogic input
6	DOUT1	Auxiliary control (open collector)



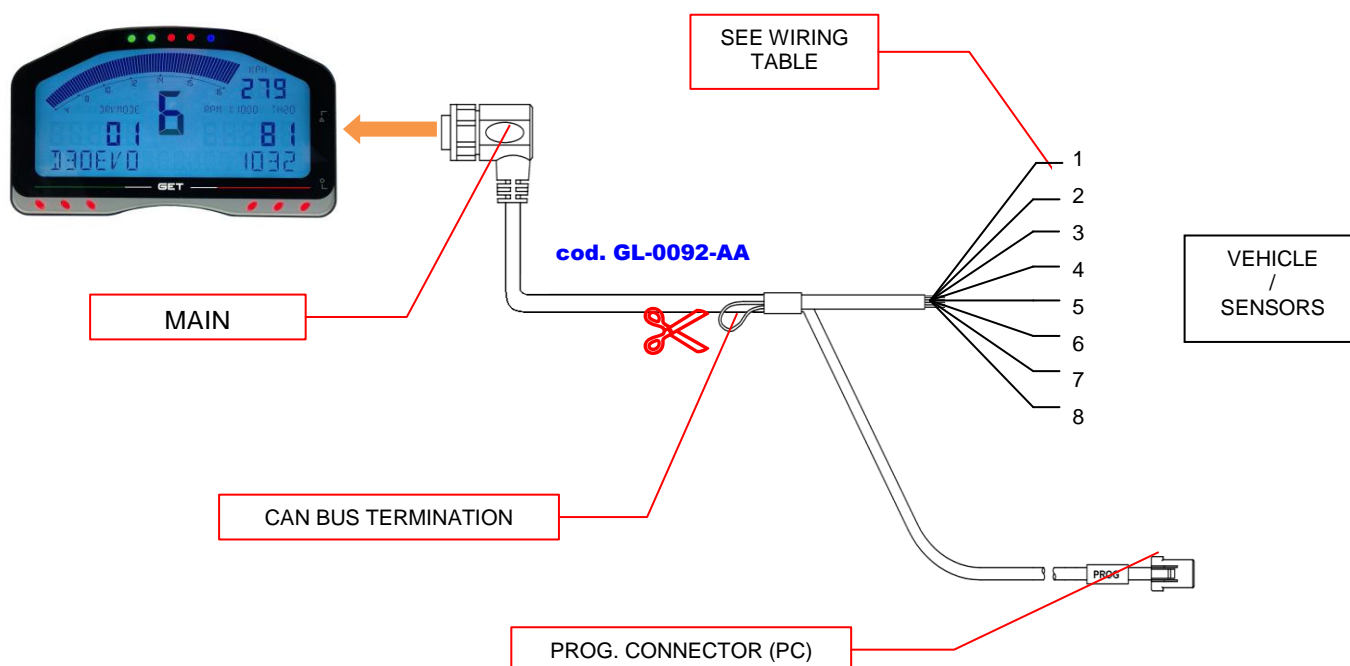
**CAUTION: THE PINOUT MAY PRESENT VARIATION IN SOME APPLICATIONS**



## 5 D30EVO: SYSTEM CONNECTION

### 5.1 Wiring code GL-0092-AA

Connection scheme with general wiring, code **GL-0092-AA**:



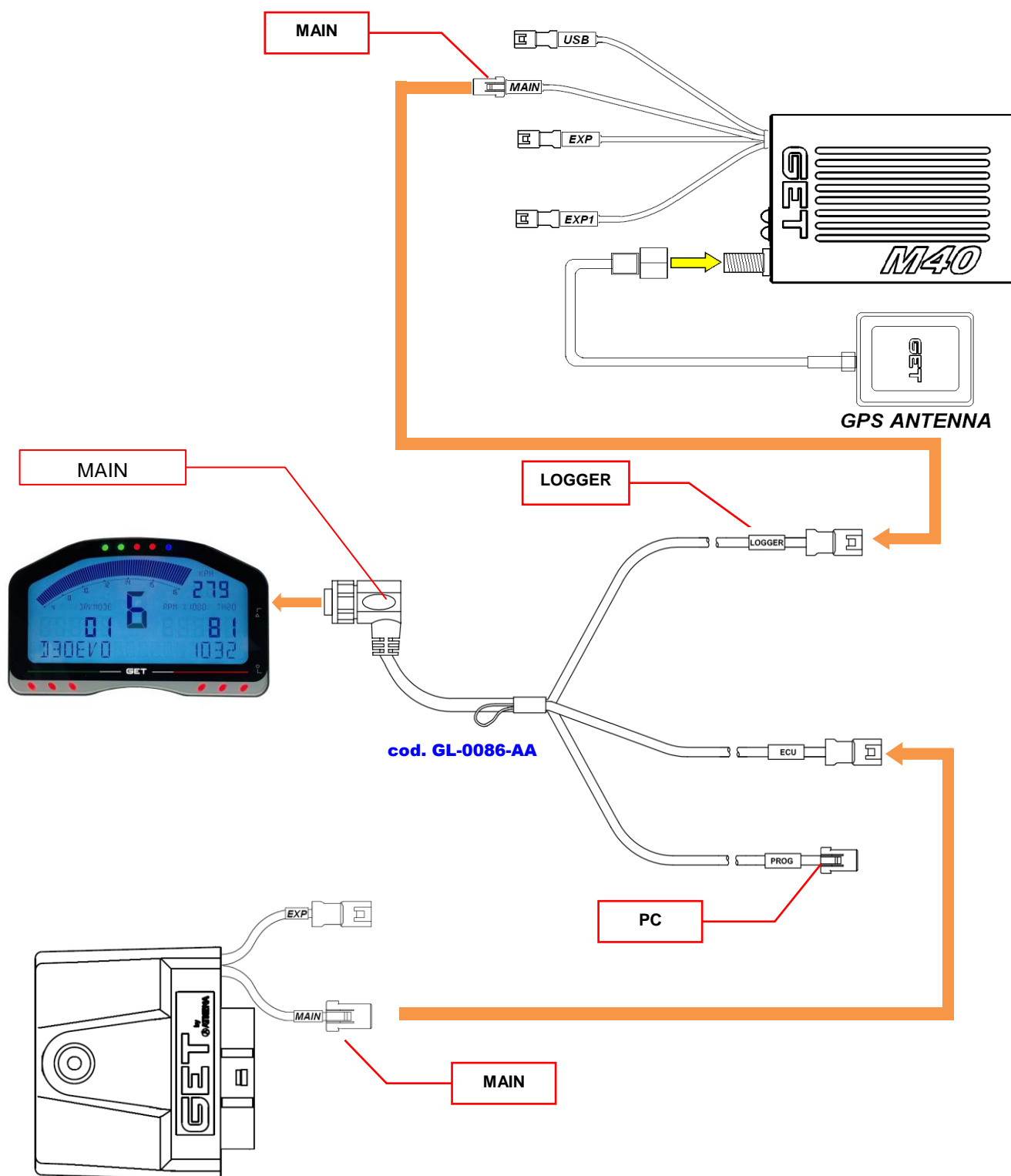
WIRING TABLE		
ITEM	COLOUR	DESCRIPTION
1	Brown	Positive supply + VBB
2	Blue	Ground supply – GND POWER
3	White	CAN bus – CANL
4	Green	CAN bus – CANH
5	Grey	Analogic input ground – GND SENSORS
6	Violet	Analogic input (0-5V) – AN1
7	Orange	Frequency input (0-12V) – DIN1
8	Light green	Ingresso Beacon – BEACON

#### NOTES:

- Cut the CAN BUS TERMINATION eyelet of the **GL-0092-AA** wiring if required for removing the internal termination of the CAN bus.
- Singularly insulate the wires that are not used.

## 5.2 Wiring code GL-0086-AA

Connection scheme with wiring, code **GL-0092-AA** for ECU GP1EVO and KM3 EVO :



### NOTES:

- In the depicted configuration, the M40 logger is supplied by the ECU

## 6 TECHNICAL CHARACTERISTICS

### *Hardware specifications:*

- Renesas RX62T Family 32 Bit RISC 100 MHz 100 MIPS microcontroller
- Memory: 32 KB Data-Flash; 16KB RAM
- Supply voltage: 13.5 VDC nominal (admitted range 9VDC – 24VDC)
- Current consumption (on nominal voltage – with no connected loads and active alerts): < 70mA
- LCD display with 1024 segments, transfective (positive FSTN)
- Serial communication port RS 232
- Programmable communication port CAN bus High Speed (1Mb/s, 500kb/s, 250kb/s, 125kb/s in format Intel or Motorola)
- Real-time clock with integrated backup battery (autonomy up to 1 month)
- Integrated internal temperature sensor
- Integrated tri-axial accelerometer (+/- 16g)
- No. 2 analogic inputs (0-5 V)
- No. 1 input for signals in frequency (up to 15 kHz)
- No. 1 input for beacon devices
- No. 1 output for open drain (maximum current 2.5 A)
- No. 1 interface connector with MAIN Amphenol 12-pole wiring
- No. 1 interface connector with AUX Amphenol 6-pole wiring

### *Miscellanea:*

- Programmable LCD backlight intensity
- 11 fully independent and programmable signalling/alert LEDs (surging channel, methods of ascension, brightness)
- 2 buttons for access to system menus
- External packaging of theft-proof and highly resistant plastic material
- Software updatable configuration
- PC updatable firmware
- Protection: IP66
- Weight (only dash): 385 g

## 7 USE OF THE D30EVO DASHBOARD

The viewed data, their layout on the display, the alerts and the number of pages are decided upon the configuration of the dashboard.

Consult the corresponding chapters for the configuration of the dashboard.

### 7.1 Functional keys

In the following the functions of the buttons of the **D30EVO** dashboard:

- Button  $\Delta$ : page change.
- Button  $\bigcirc$ : not implemented.

## 8 PRINCIPLES FOR THE CONFIGURATION OF THE D30EVO DASHBOARD

The dashboard can be set up by using the **MAYA** software; consult the corresponding user manual for the installation and use of the programme.

The configuration firstly requires the setup of the parameters of the channels to view through the **Dash Channels** map of the **Maya**.

The layout of the configured channels and of the related alerts shall be done through **Page** type maps.

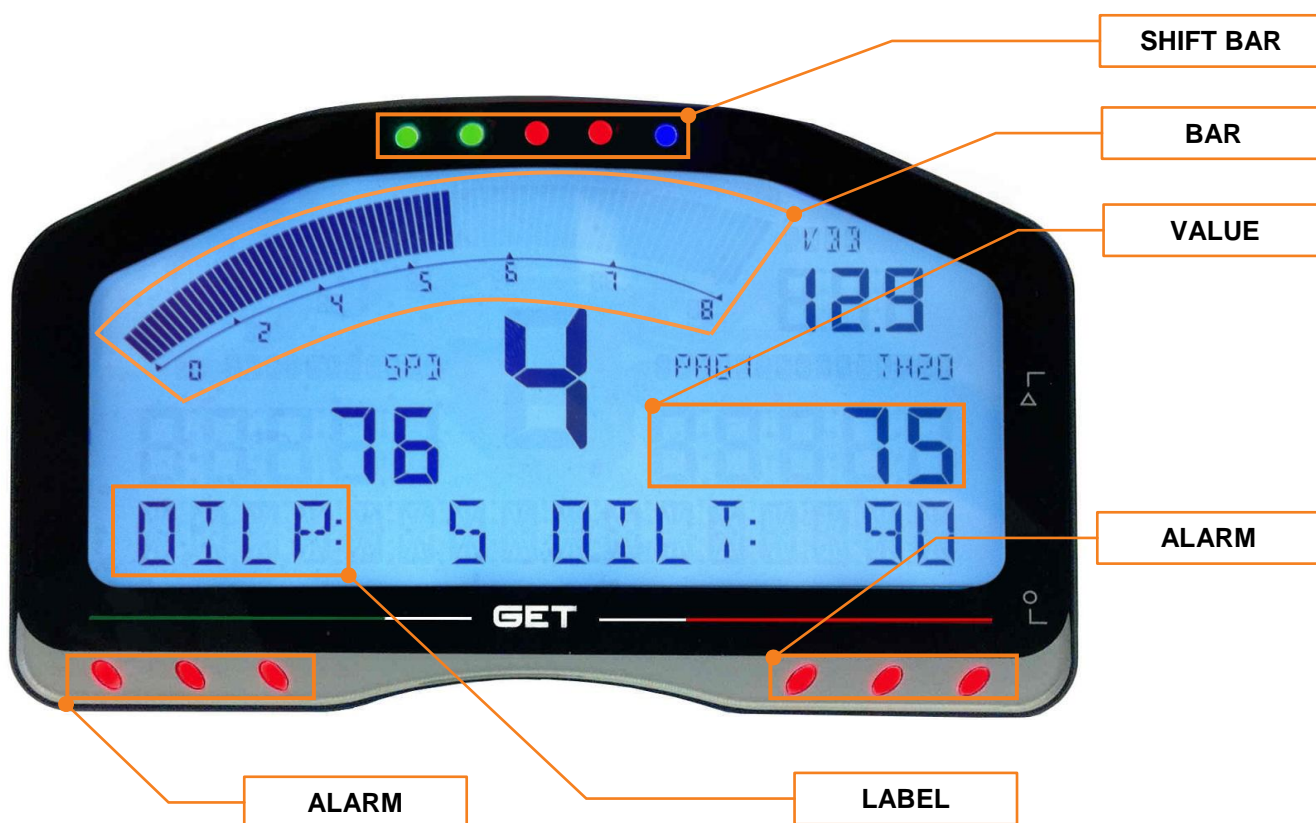
The normal layout of the dashboard foresees the use of **display fields** such as:

- **BAR** : bar of the engine routes with the corresponding scale labels
- **LABEL** : fields that only include the names of the channels viewed in **VALUE** type fields
- **VALUE** : fields devoted to the view of channel values

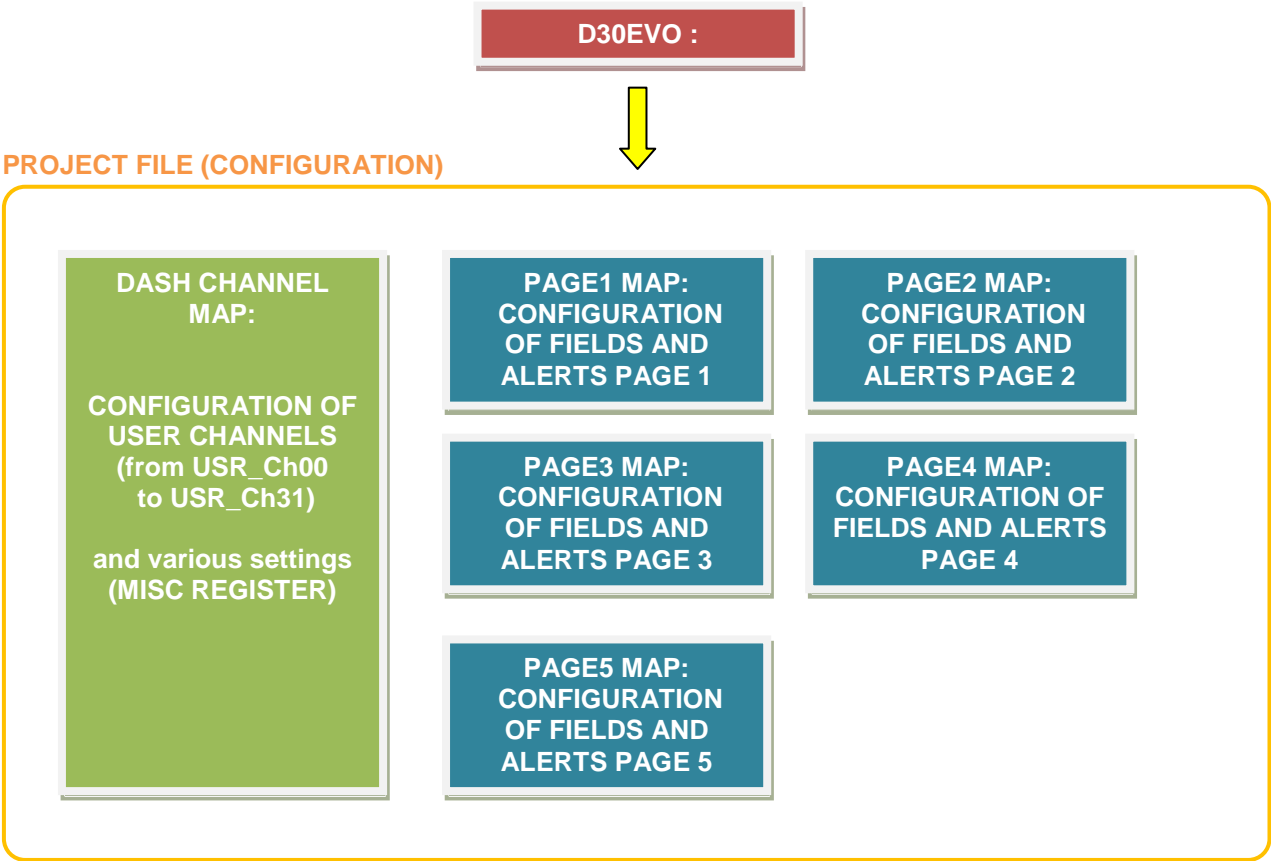
A range of alert leds is also available:

- **SHIFT BAR** : alert led for engine routes
- **ALARM** : led for various alerts

In the following, the graphical layout of the dashboard



The diagram below illustrates the software components of the **D30EVO** dashboard:



We recommend that the dashboard be configured according to the following procedure:



**NOTE: IT IS NOT REQUIRED TO CONFIGURE ALL THE PAGES OF THE DASHBOARD, WE EVEN SUGGEST THAT YOU MAY LEAVE EMPTY MAPS**

## 8.1 BAR type field

The bar of the engine routes includes two elements:

- 1 : icon of the route bar
- 2 : labels for route scale





## 8.2 LABEL type field

The **LABEL** type field is generally used for:

- identifying the value viewed in an adjacent field
- identifying the viewed page number

The figure below presents the **LABEL** type fields of the **D30EVO** dashboard:



**LABEL** type fields generally have 4 characters.

### NOTES:

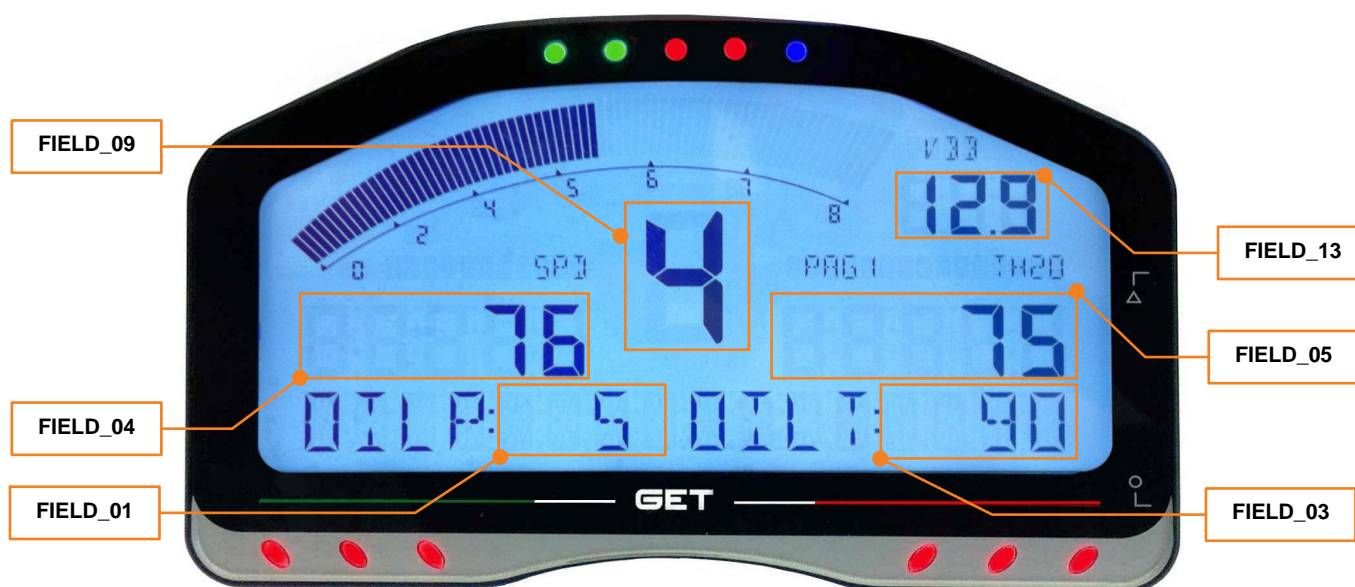
- The **FIELD\_XX** names correspond to the relative groups of the **MAYA** maps.
- Do not insert special characters (e.g.: \$, £, &, “, !, etc ... ).

### 8.3 VALUE type field

The **VALUE** type field is generally used for:

- viewing the value of a channel
- viewing particular channels (e.g. LAPTIMER on the fields **Field\_04** and **Field\_05**)

The figure below presents the **VALUE** type fields of the **D30EVO** dashboard:



The length of **VALUE** type fields is represented in the table below:

FIELD NAME	NO. OF CHARACTERS	NOTE
FIELD_01	3	except for " : " " . "
FIELD_03	4	except for " : " " . "
FIELD_04	6	except for " : " " . "
FIELD_05	6	except for " : " " . "
FIELD_09	1	
FIELD_13	3	except for " : " " . "

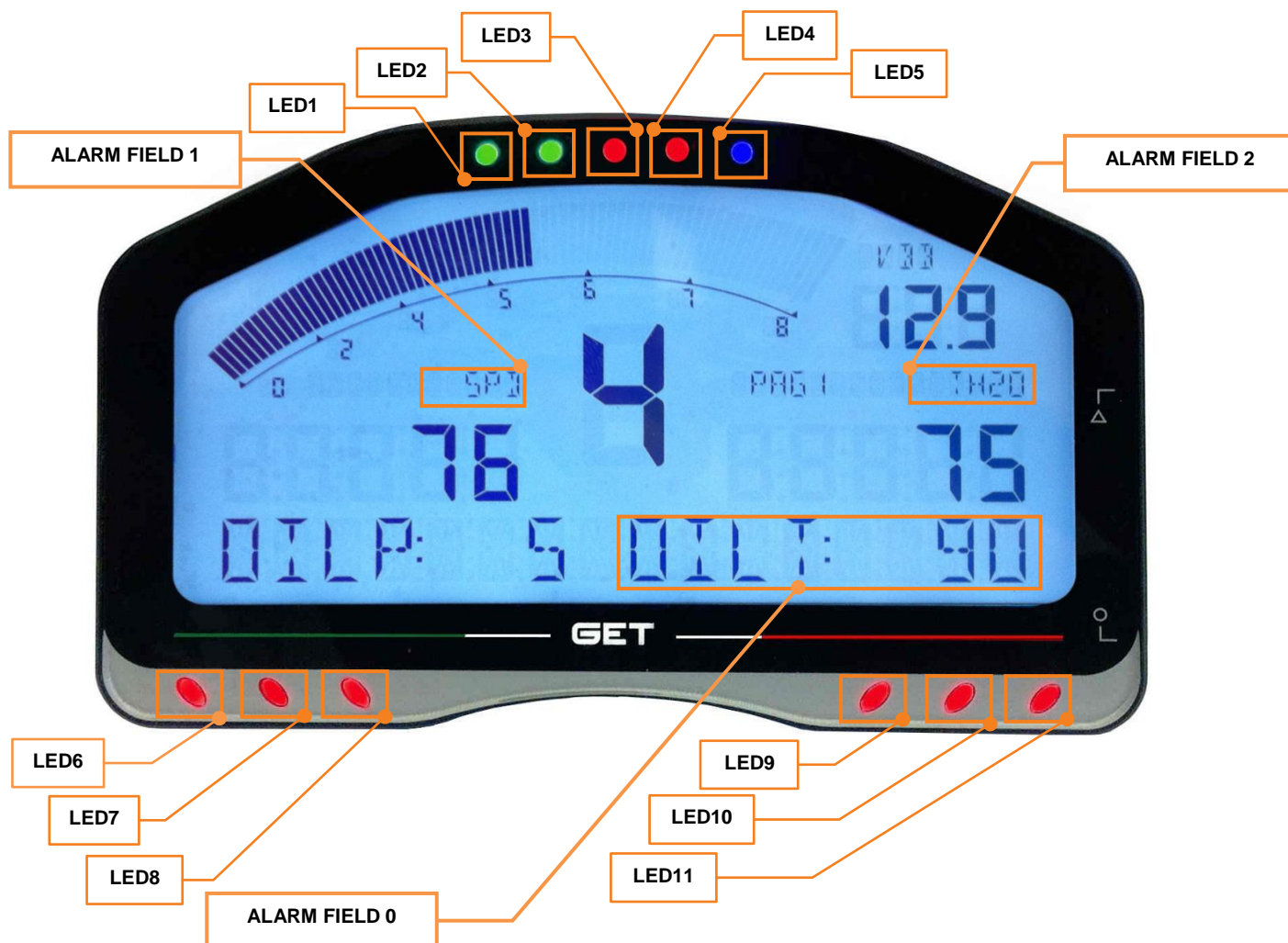
NOTE: The **FIELD\_XX** names correspond to the relative groups of the **MAYA** maps.

## 8.4 Signals (ALARM)

The alarm signals of the **D30EVO** dashboard are divided into:

- **LED...** : alarm signal
- **ALARM FIELD...** : alarm message field

The figure below presents fields of alarm LEDs:



NOTE: The names indicated in the figure correspond to the relative components of the scalars of the **MAYA** maps.

## 9 CONFIGURATION OF D30EVO: PRELIMINARY ACTIONS

Before initiating the setup of the **D30EVO** dashboard you must:

- Possess the **MAYA** software (**EVO** or **ADVANCE** license)
- Install and set up the **MAYA** software (see the software user manual)
- Possess the **Device** file of the **D30EVO** dashboard (see user manual of the **MAYA** software)
- Possess the project file of the **D30EVO** dashboard

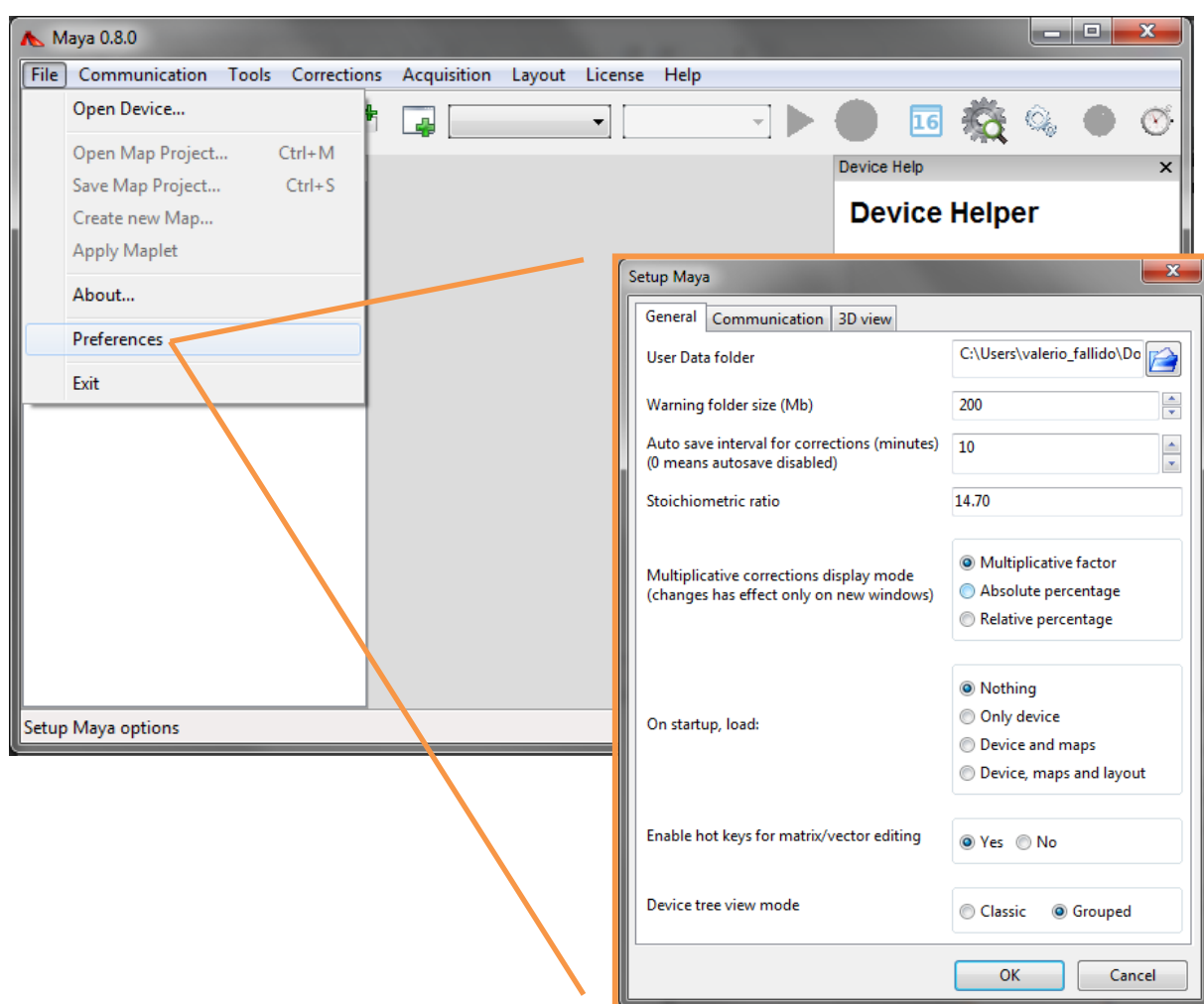
The setup files (i.e. the **Project** files of **MAYA**) may be directly downloaded from the dashboard (if the device is already setup) or may be requested to the manufacturer.

**BY ALL MEANS YOU SHOULD AVOID CREATING DASH CHANNEL MAPS FROM SCRATCH!!!**

### 9.1 Suggestions for the configuration of MAYA

You should perform the following settings/checks on the **MAYA** software:

- Set the view to **Grouped** in the software **Preferences**.
- Check the correct setup of the communication port in the software **Preferences**.



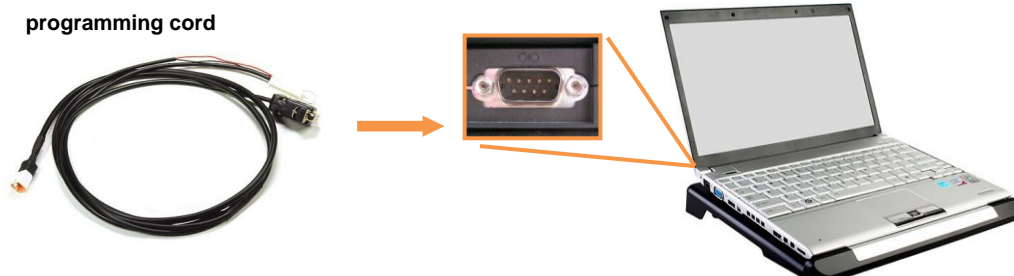
- Load the correct **Device** for the **D30EVO** device in the software.

## 9.2 Connection of the dashboard to the PC

Operate as follows:

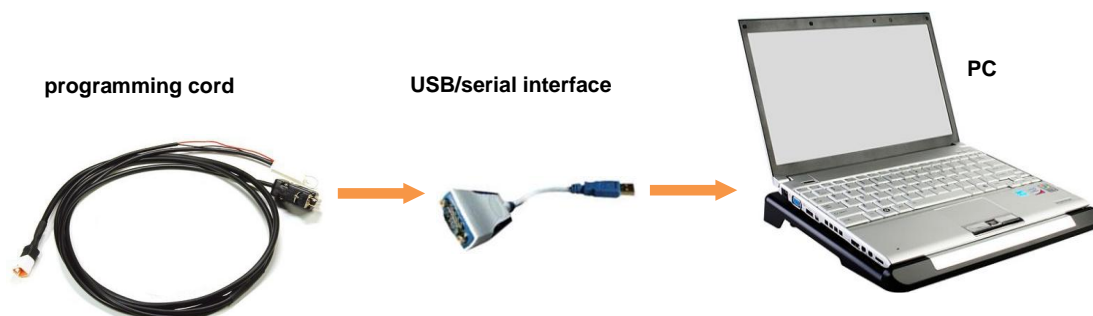
- Start the PC and the **Maya** software (if you have not done it already).
- Make sure you have correctly set up the communication port in the **MAYA Preferences**.
- Connect the programming cord to the PC (when the computer has a serial port).

### CONNECTION TO THE PC WITH A SERIAL PORT

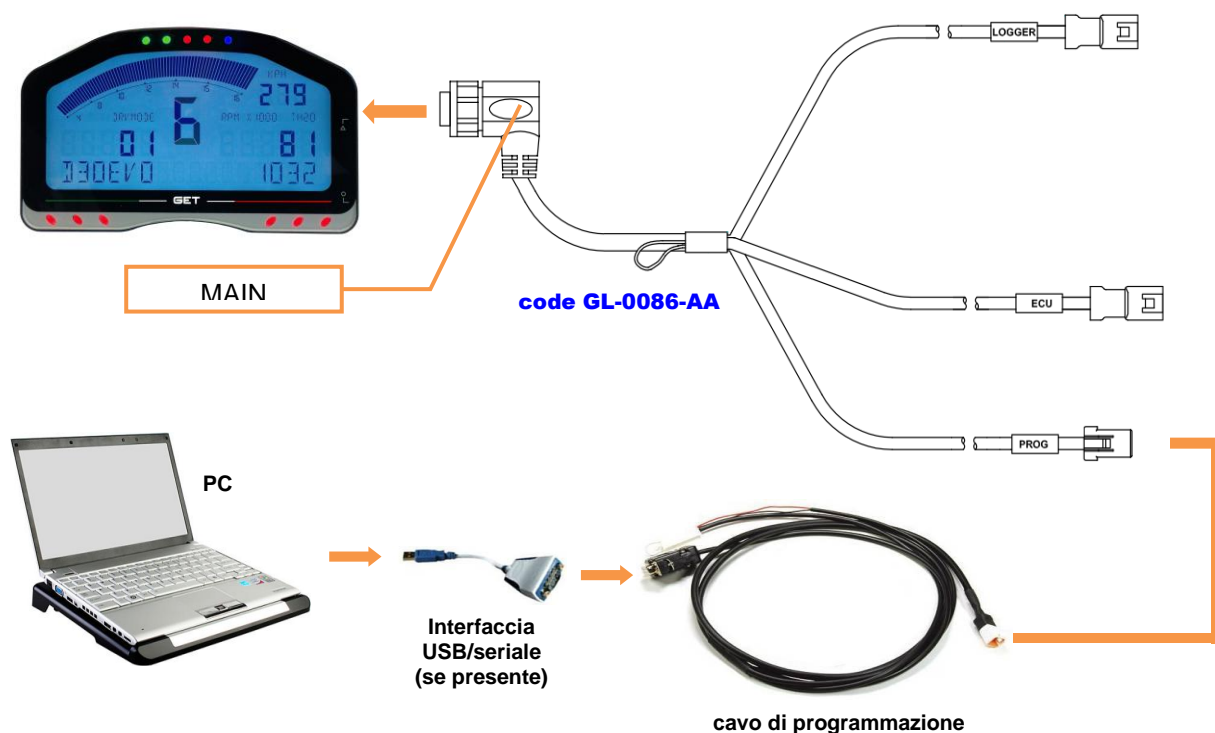


- Connect the converter (if available) to a USB port of the PC and wait for Windows© to install the driver of the new peripheral device (if required, follow the directions showed by the operating system).

### CONNECTION TO A PC WITH NO SERIAL PORT

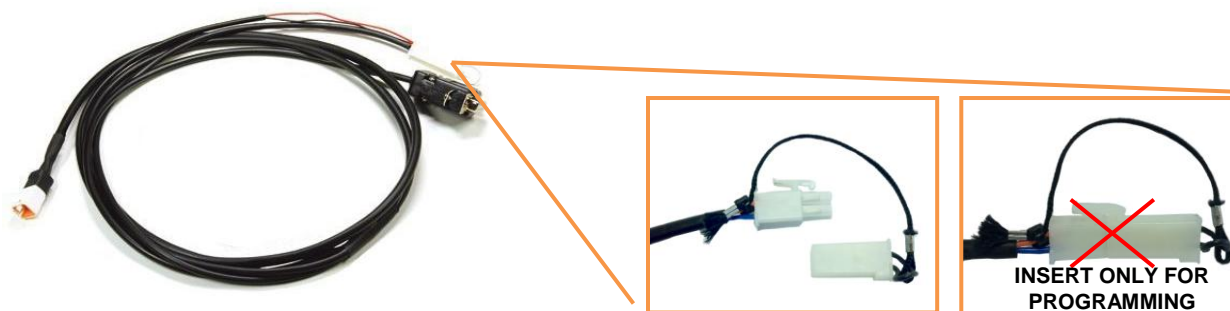


- Connect the **D30EVO** dash to the programming cord by means of the dedicated connector:

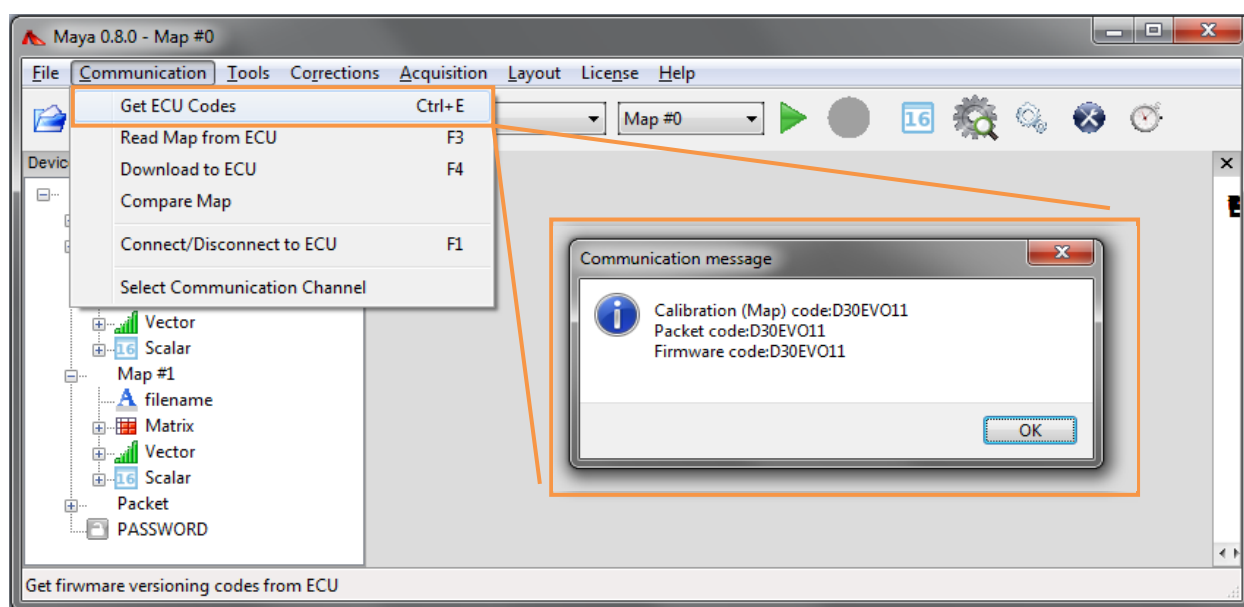




- Make sure that the programming connector (included in the cable) is disconnected (if you do not intend to programme the dashboard).



- Turn the **D30EVO** dashboard on.
- Check the correct connection of the dashboard to the PC clicking the **Get ECU Codes** control (included in the **Communication** menu). If communication is established correctly, you will see a window including the codes in the memory of the connected dashboard.





**NOTE:** this operation is not required for the connection between the dashboard and the PC, but it is useful in order to check the correct performance of cable connections and the setup of the MAYA communication port.

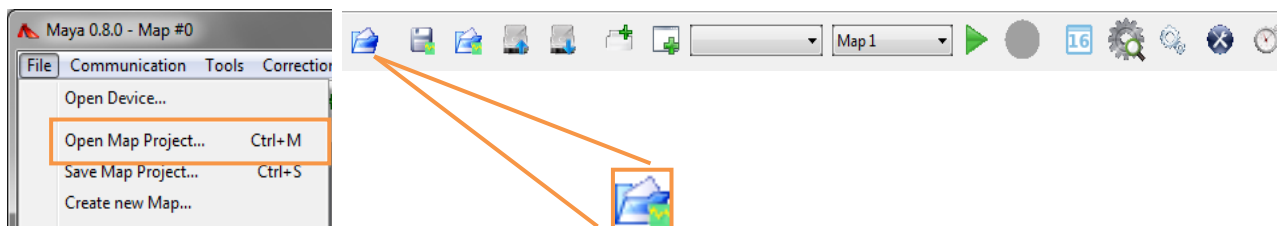
**CAUTION:** THE SHOWED CODES ARE USEFUL FOR IDENTIFYING THE DEVICE TYPE OF THE DASHBOARD CONNECTED TO THE PC.

If the dashboard device is not known, it is enough to load a random one in the Maya in order to receive some answer from the central office.

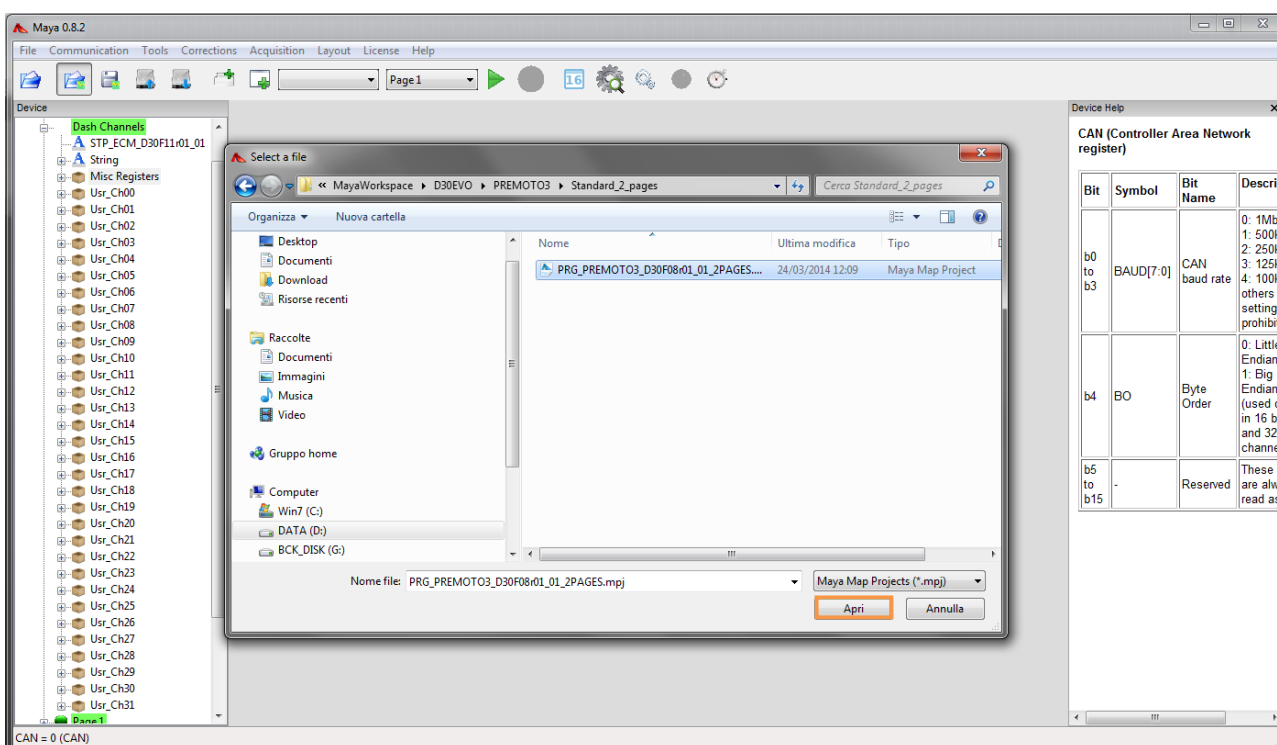
## 9.3 Opening a project (configuration) file from the PC

Operate as follows:

- Start the **Maya** software clicking the corresponding icon twice .
- Make sure that a device is loaded and that it is consistent with the project you intend to open.
- Click the **Open Map Project...** button (included in the **File** menu) or on the  icon included in the **Maya** toolbar.  
NOTE: besides the indicated methods, you may also use **Ctrl+M** on the keyboard (enabling the option **Enable Hot keys...** in the **Maya Preferences**) in order to recall the function.



- Select the desired project file. If the installation procedure is followed correctly, thus creating the **MayaWorkspace** folder, the file should be included in the folder corresponding to the configuration type included in the dashboard. For instance, a configuration for **PREMOTO3** will be saved in:  
**MayaWorkspace \ D30EVO \ PREMOTO3**




- Click the **Open** button at the bottom left in order to load the selected file
- All the engine maps included in the project will be loaded in the **device** tree (visible in the area of **Device Manager**); moreover, the engine map (identified by the green indicator of the tree) will also be enabled.




## 9.4 Downloading the configuration of the dashboard

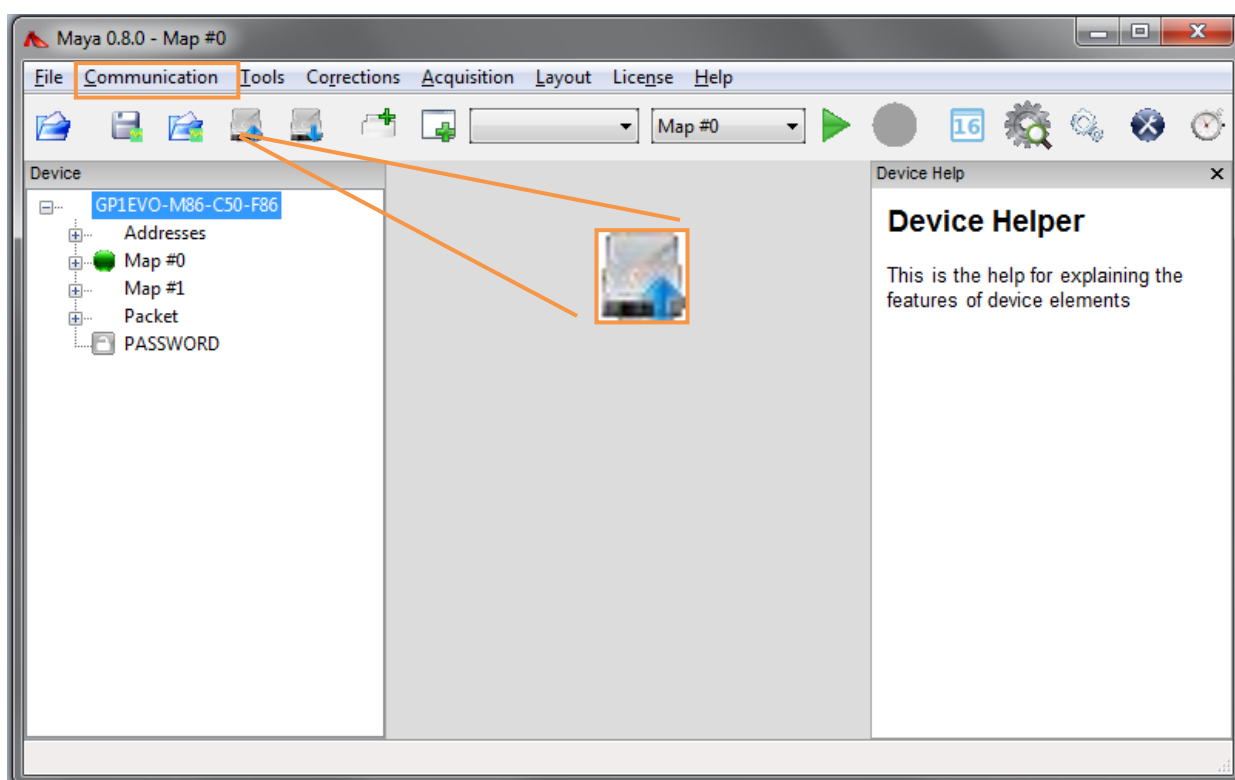
Operate as follows:

- Start the **Maya** software clicking the corresponding icon twice 
- Make sure that the device is connected to the PC.
- Make sure that the **device** viewed in the area of the **Device Manager** is consistent with the map you intend to download. For this purpose, it may be useful to perform the command **Get ECU Codes**, that helps identify the codes in the dashboard.

NOTE: if no device has been loaded, load one.

- Click the **Read Map from ECU** button (included in the **Communication** menu) or on the  icon included in the **Maya** toolbar.

NOTE: besides the indicated methods, you may also use **F3** functional key on the keyboard (enabling the option **Enable Hot keys...** in the **Maya Preferences**) in order to recall the function.



- Afterwards, the **Maya** reading command will ask you to select an engine map. The options (selectable through a click of the left key of the mouse) are the following:

**Application:** the reading of the **Application Map** saved in the device will be launched – the downloaded data may be viewed (and afterwards changed) within the **Application** tree of the **Device Manager** of the **Maya**.

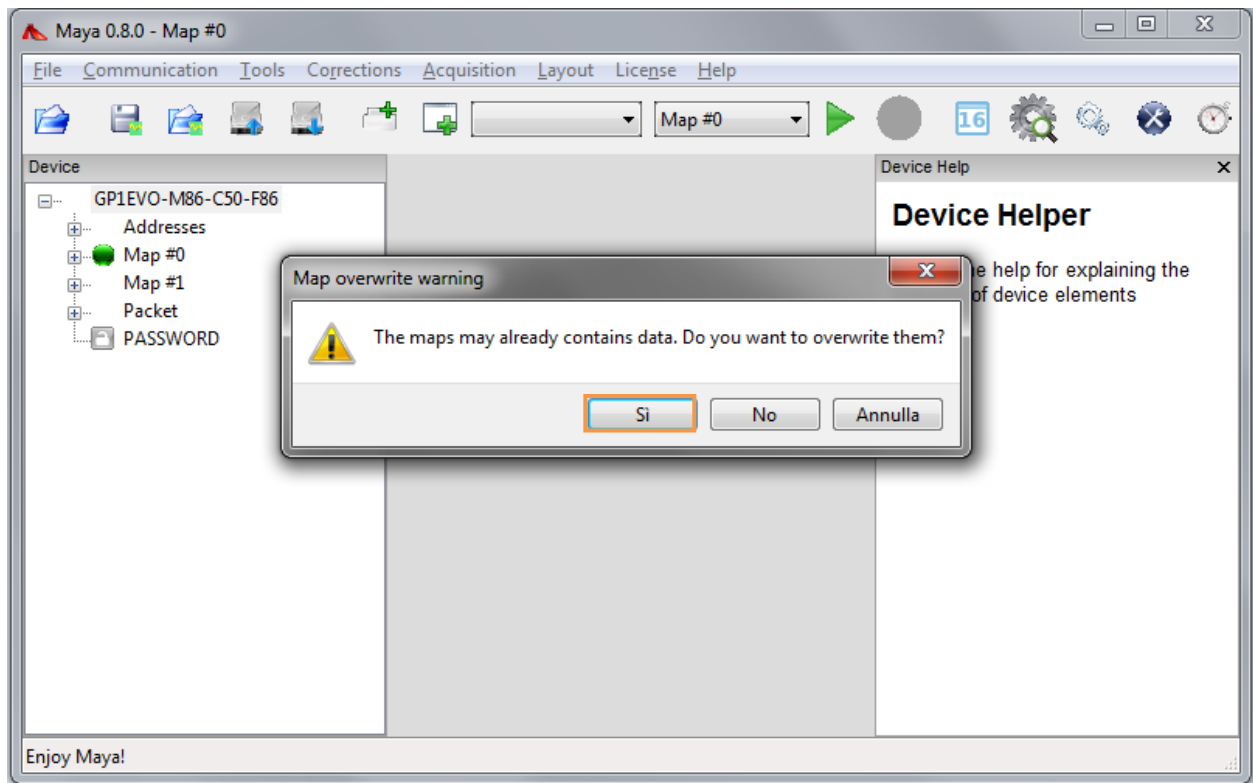
**Map ...:** the reading of the selected map will be launched – the downloaded map may be viewed (and changed) in the **device** tree.

**All:** the reading of all the maps included in the device memory will be launched – the downloaded maps may be viewed (and changed) in the **Device Manager** tree of the **Maya**.

**CAUTION: THE NUMBER OF MAPS OR THEIR NUMBERING MAY VARY ACCORDING TO THE CONNECTED DEVICE**

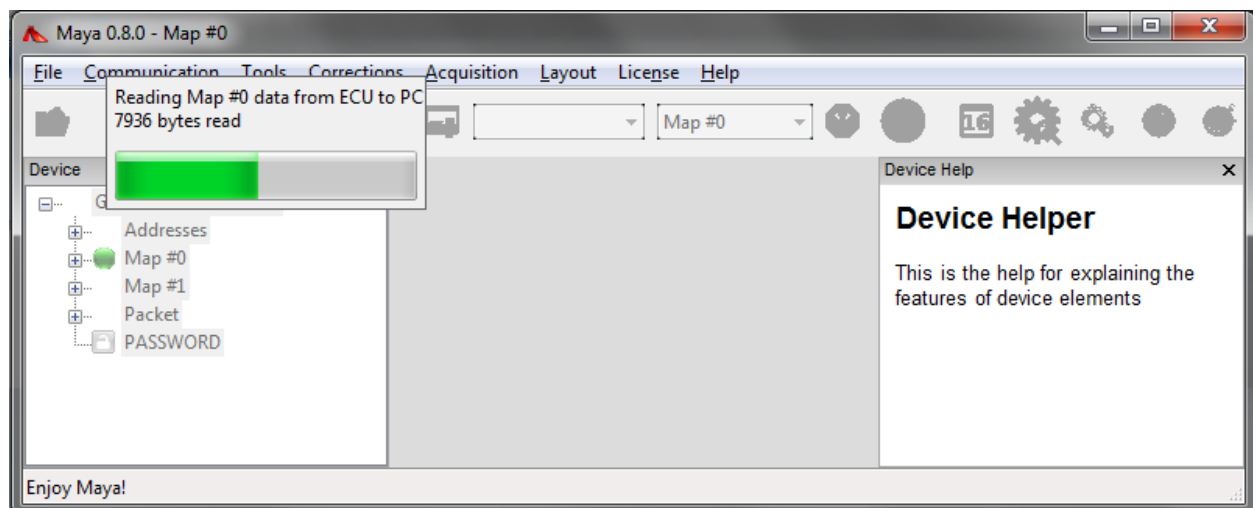
**ALWAYS SELECT THE “ALL” OPTION**

**CAUTION: IN CASE A MAP HAS PREVIOUSLY BEEN LOADED IN THE SAME POSITION OF THE ONE YOU INTEND TO DOWNLOAD, MAYA WILL WARN THE USER WITH THE MESSAGE VIEWED IN THE FOLLOWING FIGURE.**



**NOTE:** If you wish to overwrite the data, press Yes.

- Confirm the selection with the button **Ok** (bottom right of the map selection window): the reading process begins. Wait until the operation is completed.



- The downloaded engine map is loaded in the **device** tree (that can be viewed in the **Device Manager** area).

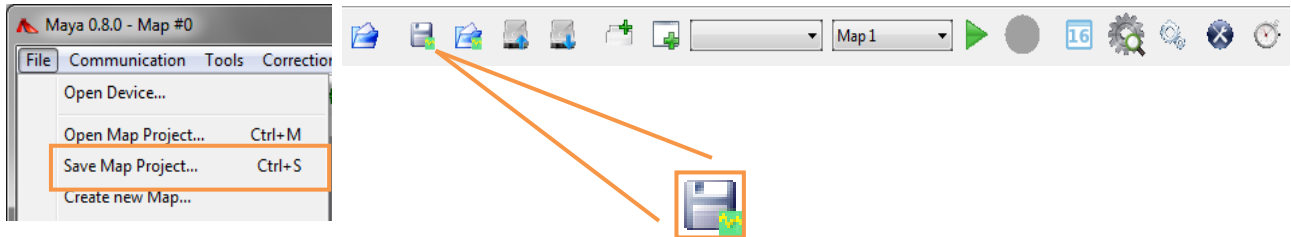
**NOTE:** When a map has been loaded, the characters **MAP # ...** is outlined in green

## 9.5 Saving a project

After having changed a map (or after any change) you should save the new content in a **project** file in your PC (or on any other data storage device).  
The procedure described below implies that the **Maya** is started on the device and the maps are already loaded.

Operate as follows:

- Click the **Save Map Project...** button (included in the **File** menu) or on the icon included in the **Maya** toolbar.  
NOTE: besides the indicated methods, you may also use **Ctrl+S** on the keyboard (enabling the option **Enable Hot keys...** in the **Maya Preferences**) in order to recall the function.




- Select the folder and input the file name of the project, confirm the save clicking the **Save** button.  
NOTE: for consistency purposes, we recommend that you should maintain the saving folder inside the **MayaWorkspace**.

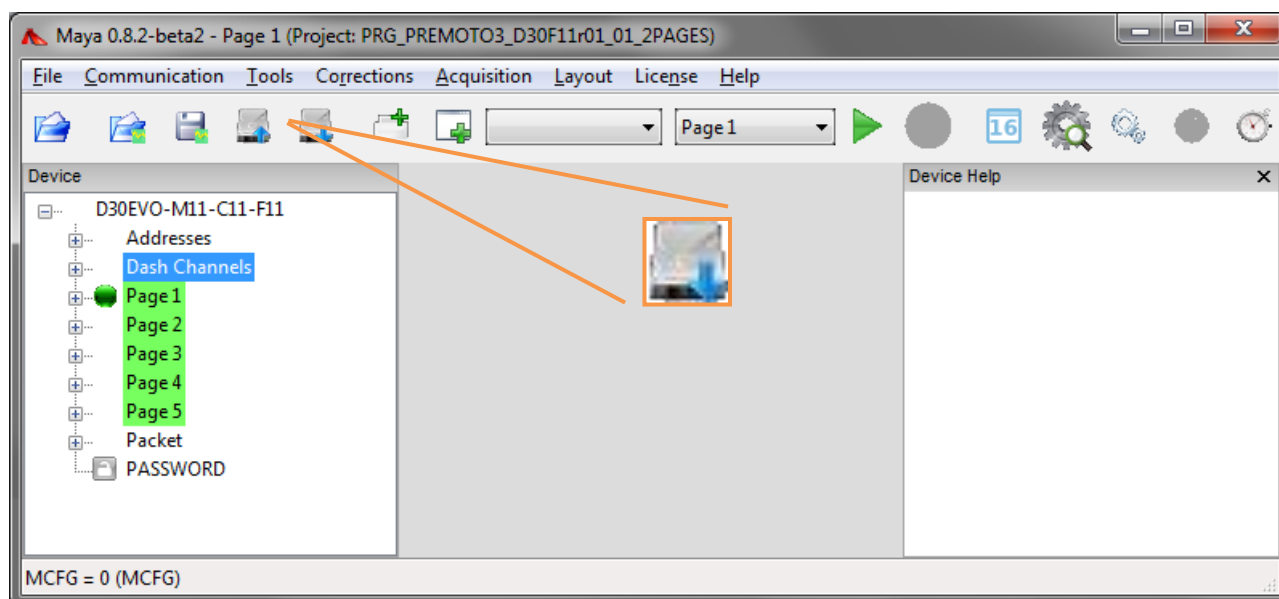
**WITH THIS PROCEDURE, ALL THE MAPS LOADED IN THE MAYA ARE SAVED IN A SINGLE FILE, WHICH EASES THEIR MANAGEMENT**

## 9.6 Programming the D30EVO:

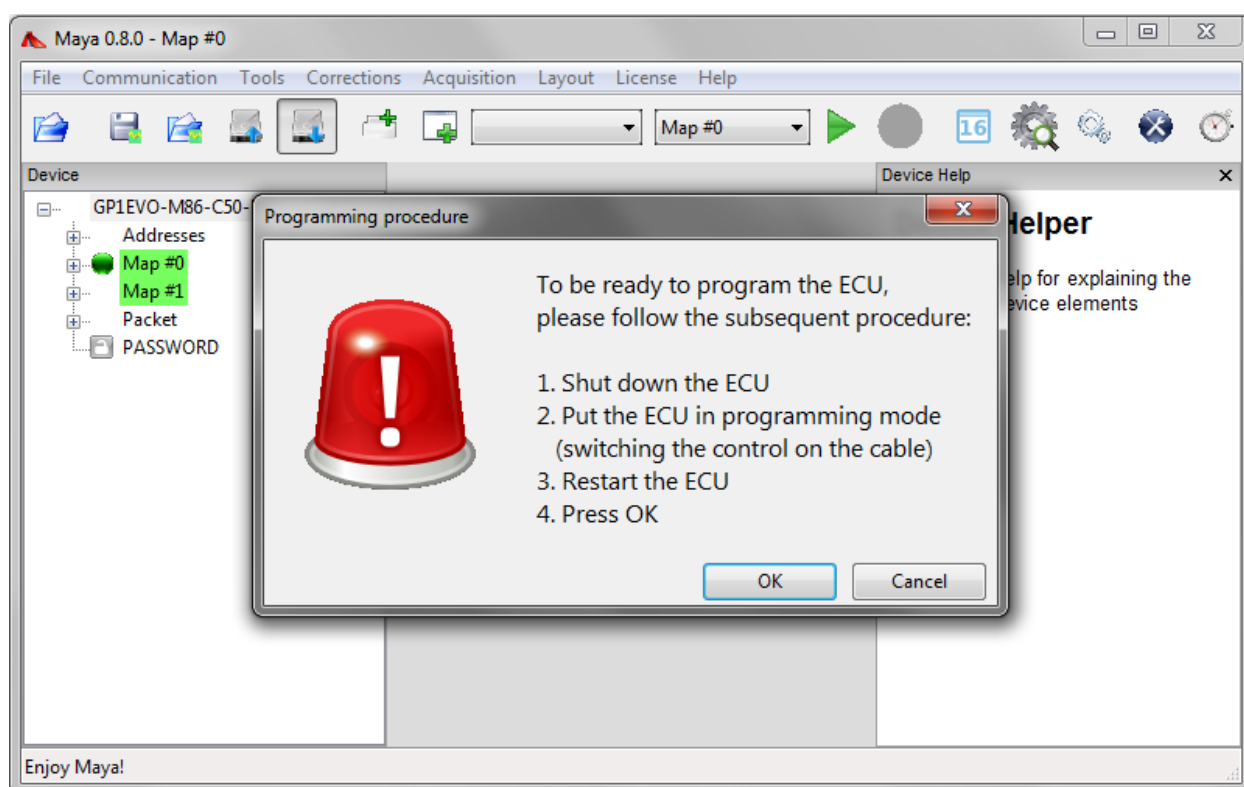
The procedure described below implies that the **Maya** is started on the device and the project (configuration) is already loaded.

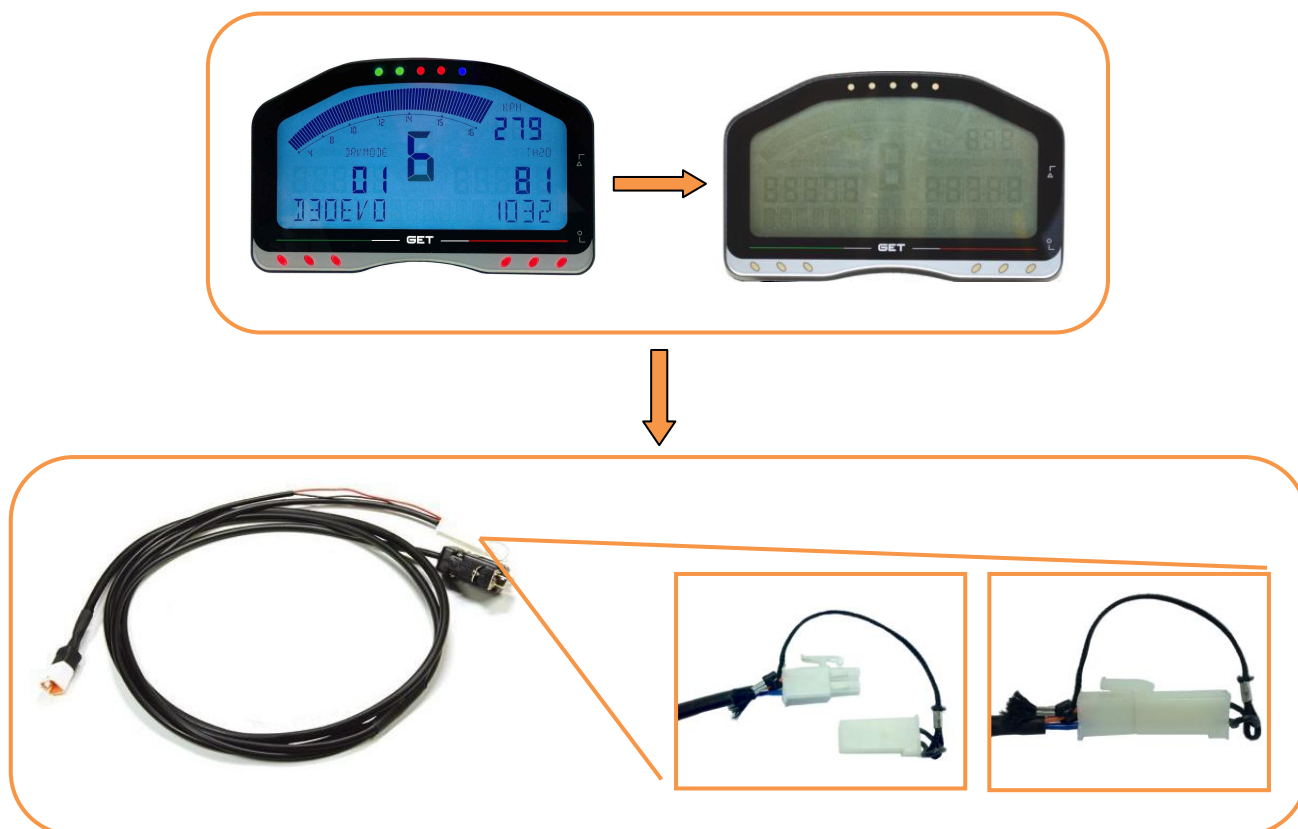
Operate as follows:

- Make sure that the device is connected to the PC.
- Click the **Download to ECU** button (included in the **Communication** menu) or on the  icon included in the **Maya** toolbar.  
NOTE: besides the indicated methods, you may also use **F4** functional key on the keyboard (enabling the option **Enable Hot keys...** in the **Maya Preferences**) in order to recall the function.

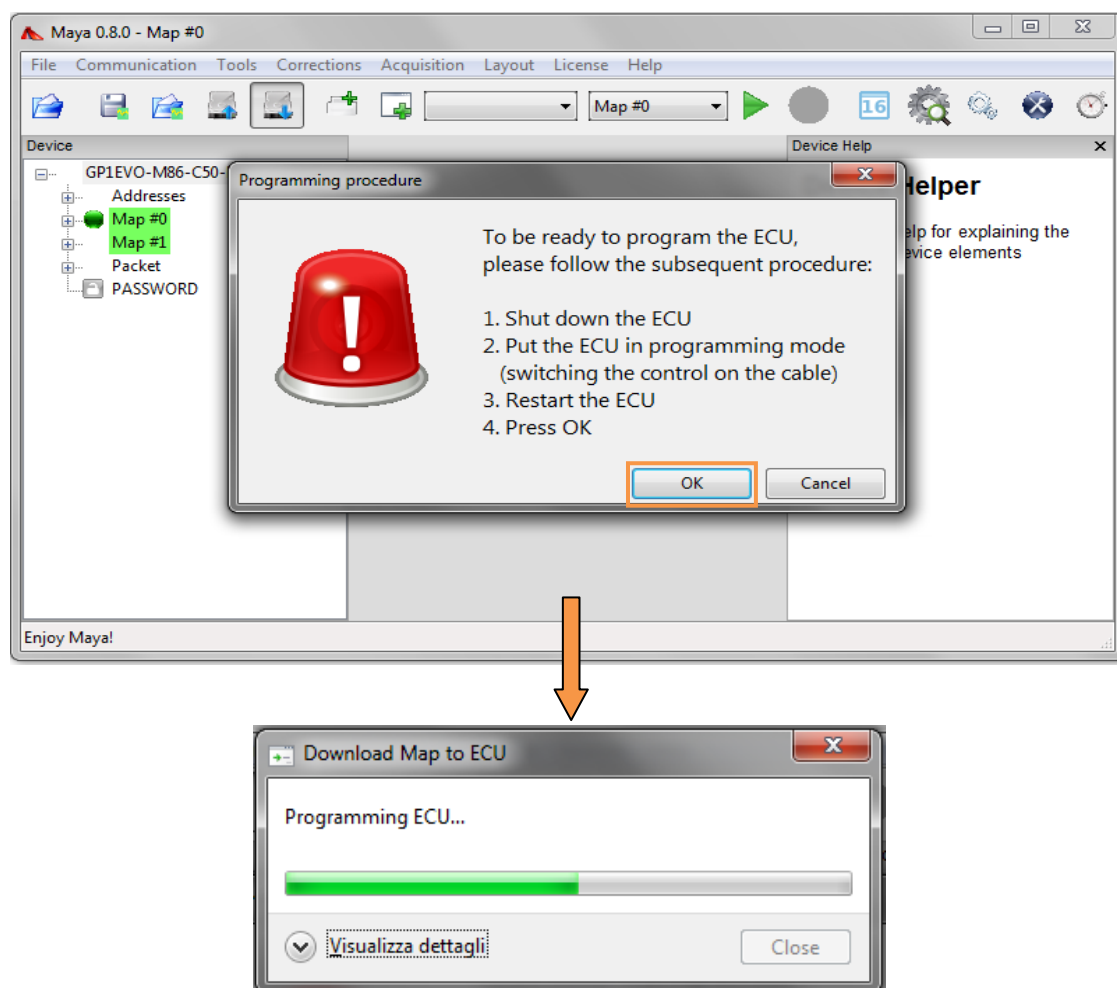


- Program the device connected to the PC following the instructions showed by the message of the **Maya** software.

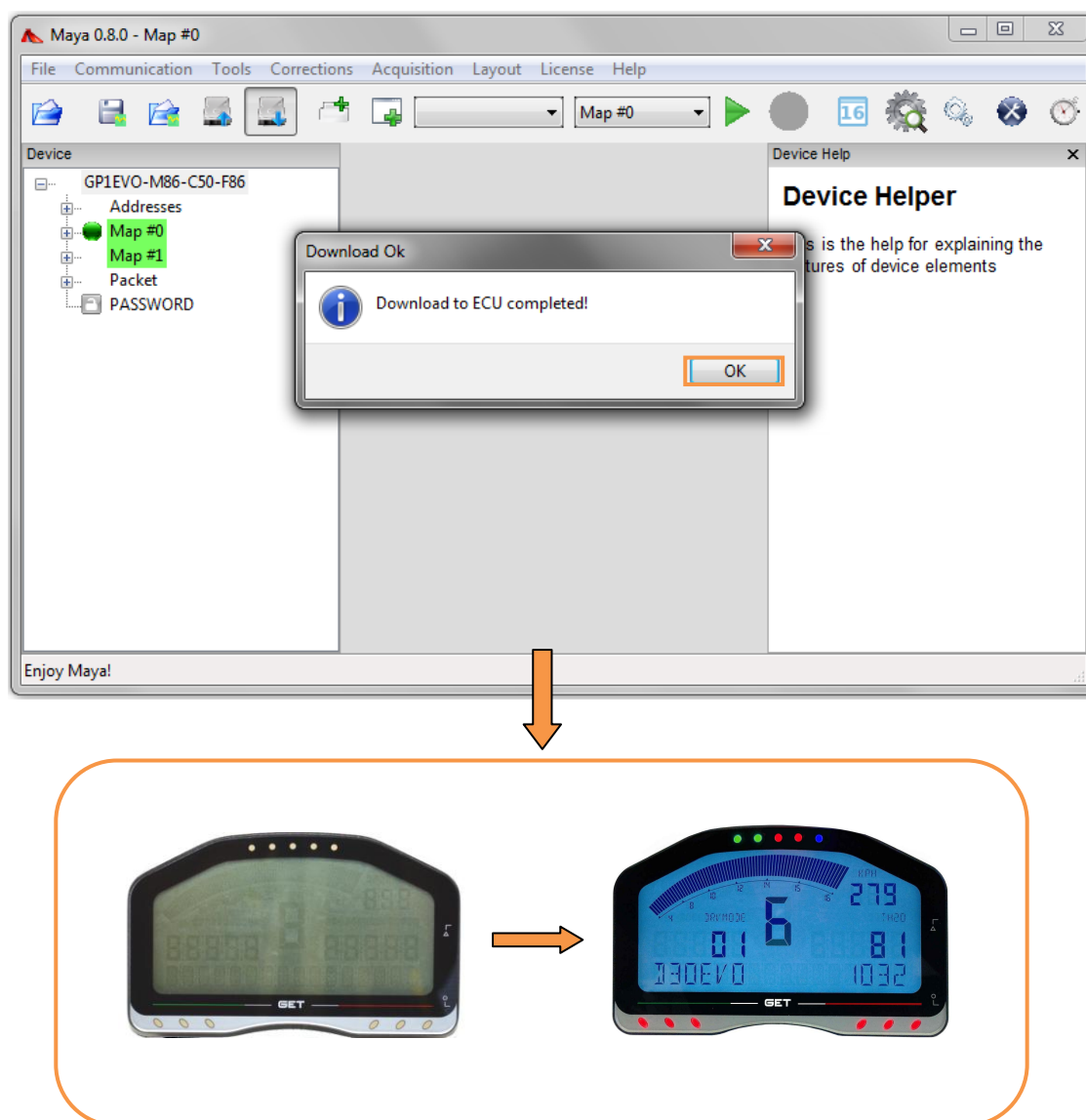




- Press the **Ok** button in order to start programming and wait until the data transfer is completed.



- When the programming ends, turn the device off and remove the programming connector



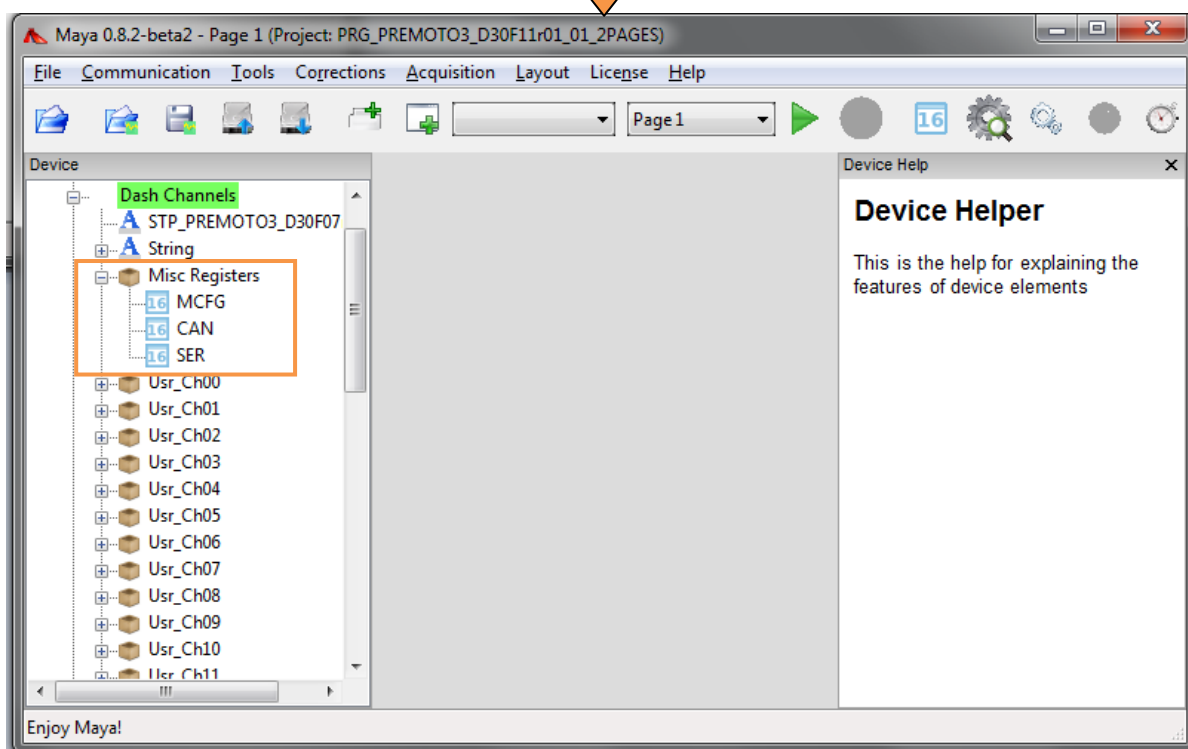
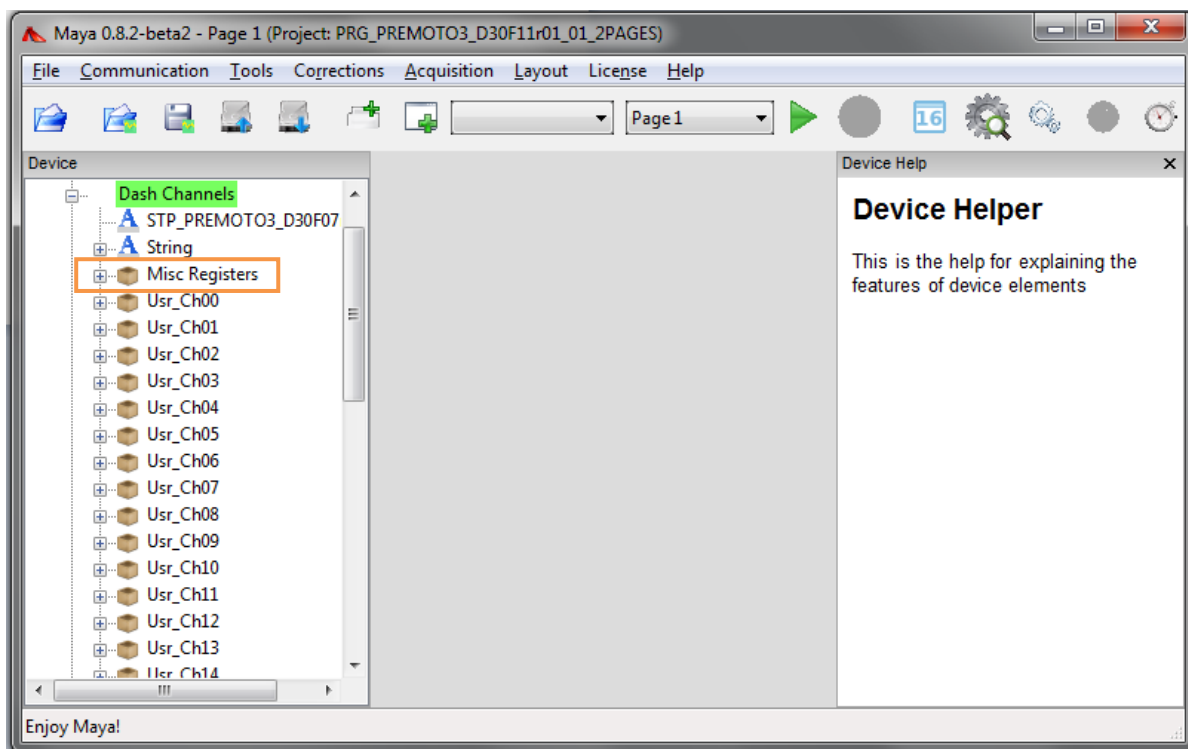
**NOTE:** in case of errors in the process, check the correct connection of the device to the PC, the position of the programming connector and the correct setup of the communication port.

## 10 CONFIGURATION OF D30EVO

Before changing maps, please read the manual of the **MAYA** software: many of the concepts expressed in such document are actually omitted in this manual.

The following chapters list the available groups for the **ADVANCE** and **EVO** licenses, for the scalars in this content please see the scalars available in the **DEVICE TABLE** of the hereby manual.

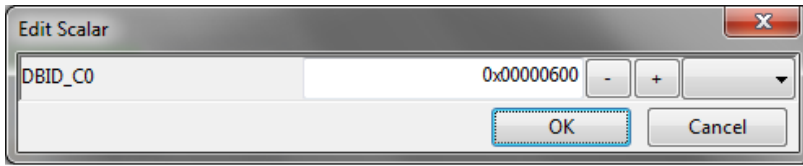
Parameters can be selected by expanding the group you intend to change and by clicking the desired scalar twice





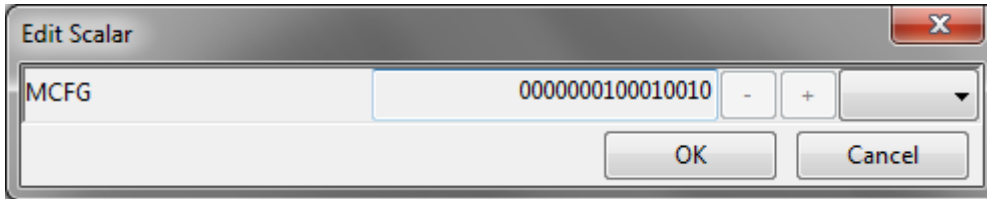
Remember that you have to double click the values of the **Scalars** included in the **Groups** in order to change them: verrà aperta una finestra per l'editor dei valori.

The scalars may be of type **Value** or they may be variables that include a value:

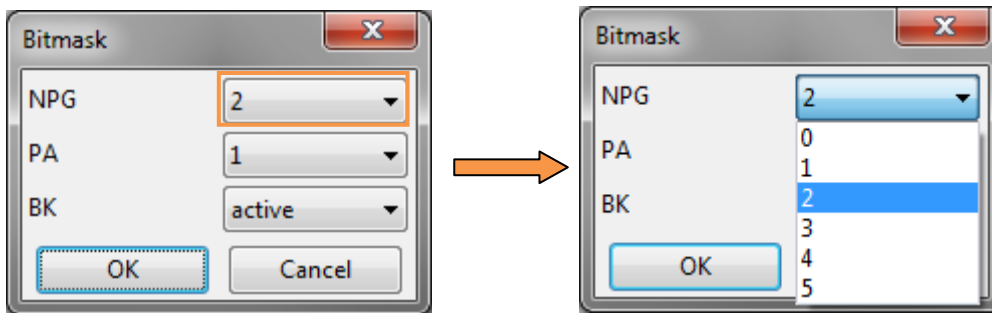


This type of scalars may be modifying by double clicking the box including the value and inputting the new data.

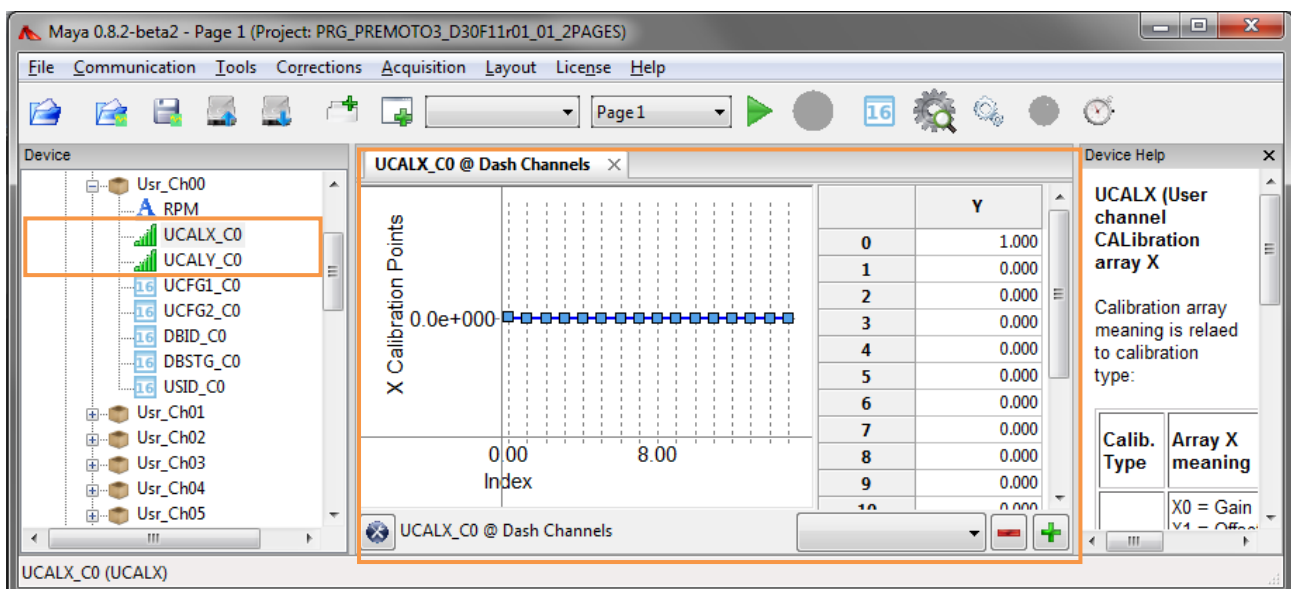
**Bitmask** scalars are variables that include a range of zeroes and ones:



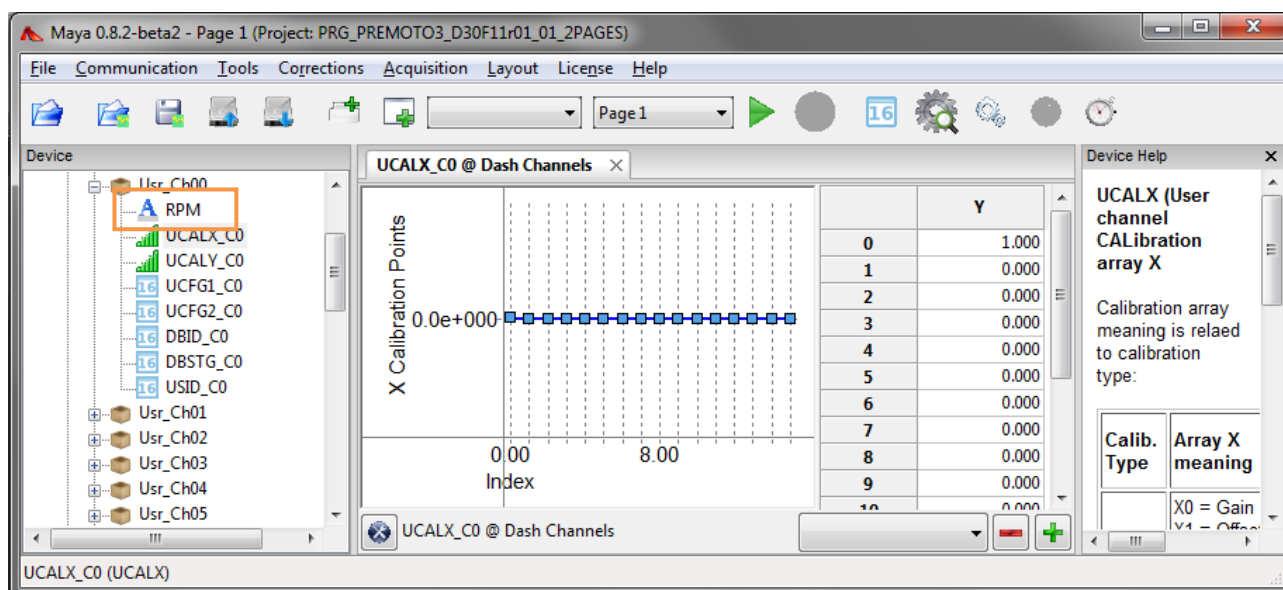
In this case, values are always changed by double clicking the mouse, but, unlike **Value** type scalars, a subsequent window will be opened so that you may select the desired value:



The vectors (**Vector**) for the possible linearization of the data received and/or showed by the dashboard are also present besides scalars.



Finally the **String** component acts as a string for recognising the selected channel and/or the name viewed on the dashboard (in **Label** type fields).



## 10.1 Dash Channel Map

The groups present in the **DASH CHANNEL MAP** are the following:

- **Misc Register:** allows to setup the number of viewed pages, the initial page, turn on the backlight, setup the speed and byte order of the CAN, setup the serial.
- **Usr\_Ch00...Usr\_Ch31:** allow to setup the parameters of the user channels (channel name, CAN parameters, calibration, etc.) to be viewed on the dashboard.

NOTE: consult the **DEVICE TABLE** for details on scalars.

## 10.2 Page1...Page5 Map

The **PAGE** maps allow to view the channels setup in the **Dash Channel Map**.

The groups present in the **DASH CHANNEL MAP** are the following:

- **Misc Register:** allows to setup the number of viewed pages, the initial page, turn on the backlight, setup the speed and byte order of the CAN, setup the serial.
- **Usr\_Ch00...Usr\_Ch31:** allow to setup the parameters of the user channels (channel name, CAN parameters, calibration, etc.) to be viewed on the dashboard.

NOTE: consult the **DEVICE TABLE** for details on scalars.

## 11 HOW TO...

In the following you will find the most frequent operations of the **D30EVO** dashboard.

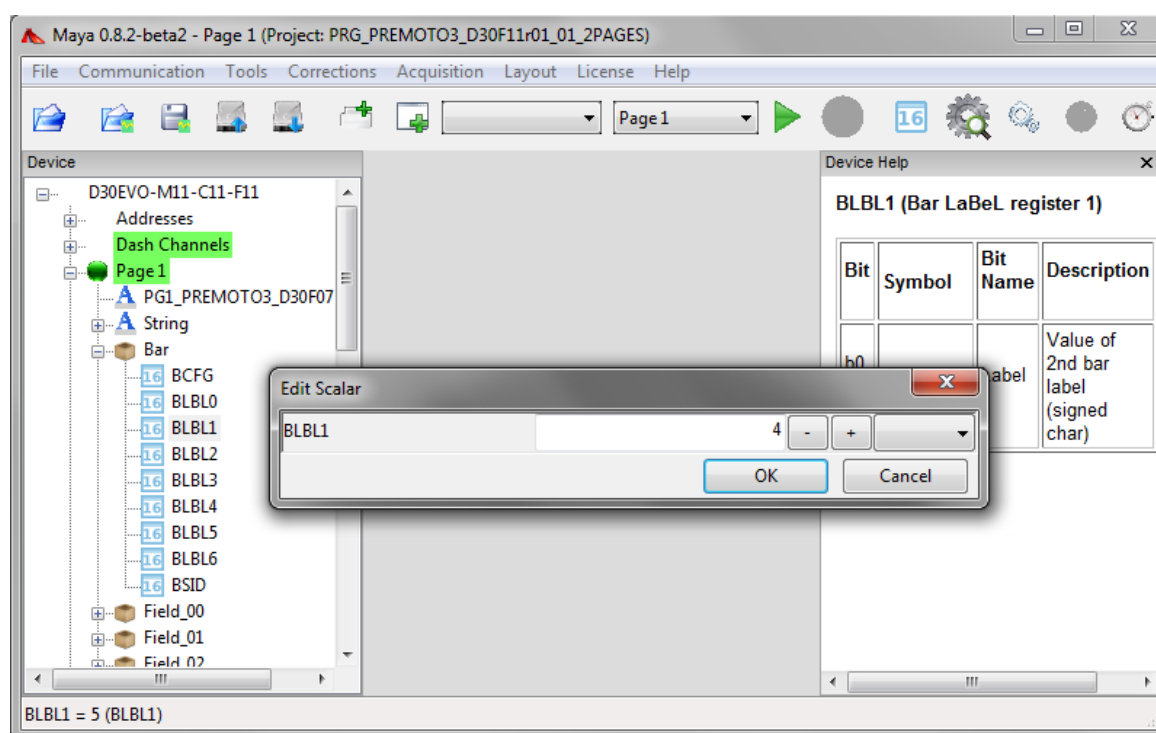
### 11.1 Changing the scale of the motor routes

In order to change the scale of the motor routes, you should change the values of the scalars in the table:

D30EVO – Page1 ... Page5 Map					
GROUP	Name	Type	Description	Lic. EVO	Lic. ADV.
Bar	BLBL0 ... BLBL6	Value	Allow to setup the 7 values of the scale of the engine routes, from the minimum one (BLBL0) to the maximum one (BLBL6)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

NOTE: the values may be changed independently in any page

The figure illustrates the value of the second label of the route scale (corresponding to 4000 rpm), setup in the **Bar** group of the **Page1** map

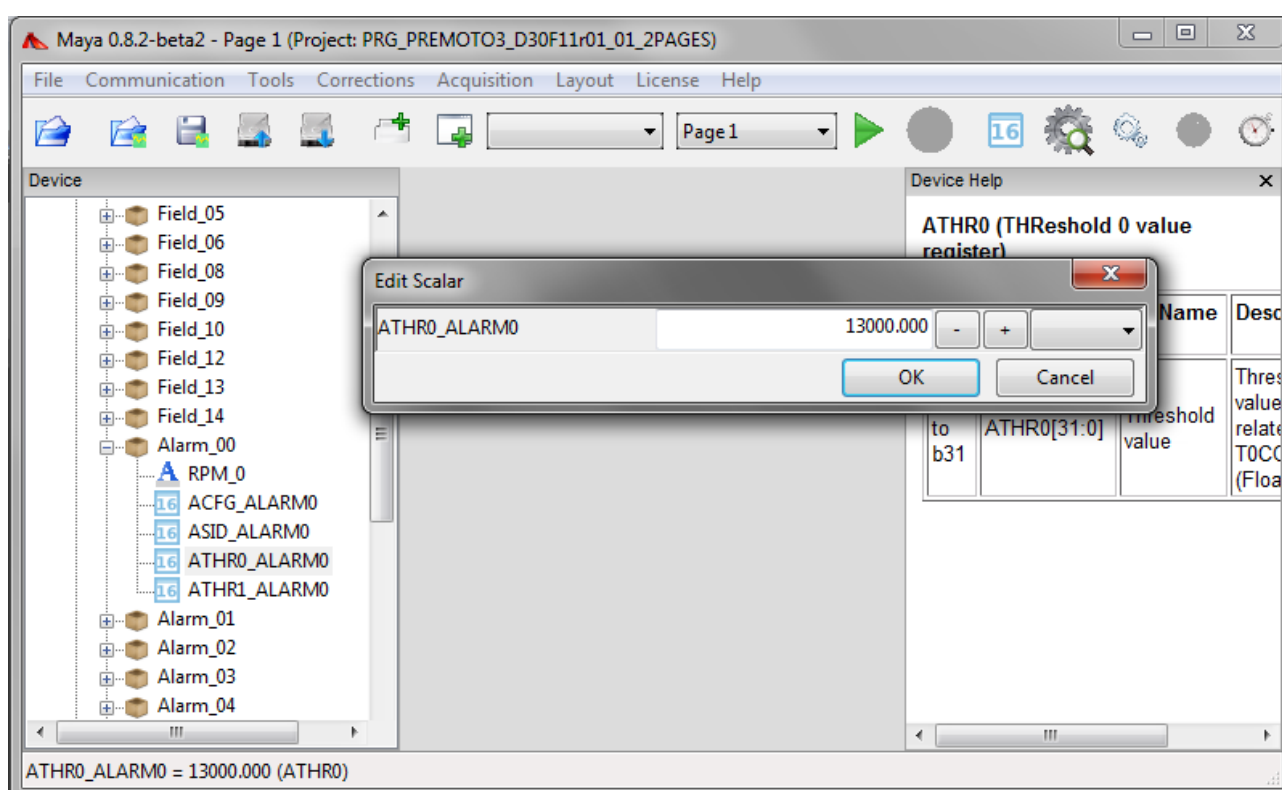


## 11.2 Changing the change flash

In order to change the change alerts, you should change the values of the scalars in the table:

D30EVO – Page1 ... Page5 Map					
GROUP	Name	Type	Description	Lic. EVO	Lic. ADV.
Alarm_0	ATHR0_ALARM	Value	Alarm limit LED1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Alarm_1	ATHR0_ALARM	Value	Alarm limit LED2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Alarm_2	ATHR0_ALARM	Value	Alarm limit LED3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Alarm_3	ATHR0_ALARM	Value	Alarm limit LED4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Alarm_4	ATHR0_ALARM	Value	Alarm limit LED5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

The figure illustrates the alarm limit (13000 rpm) for change (**RPM**) configured on the **Alarm\_01** group



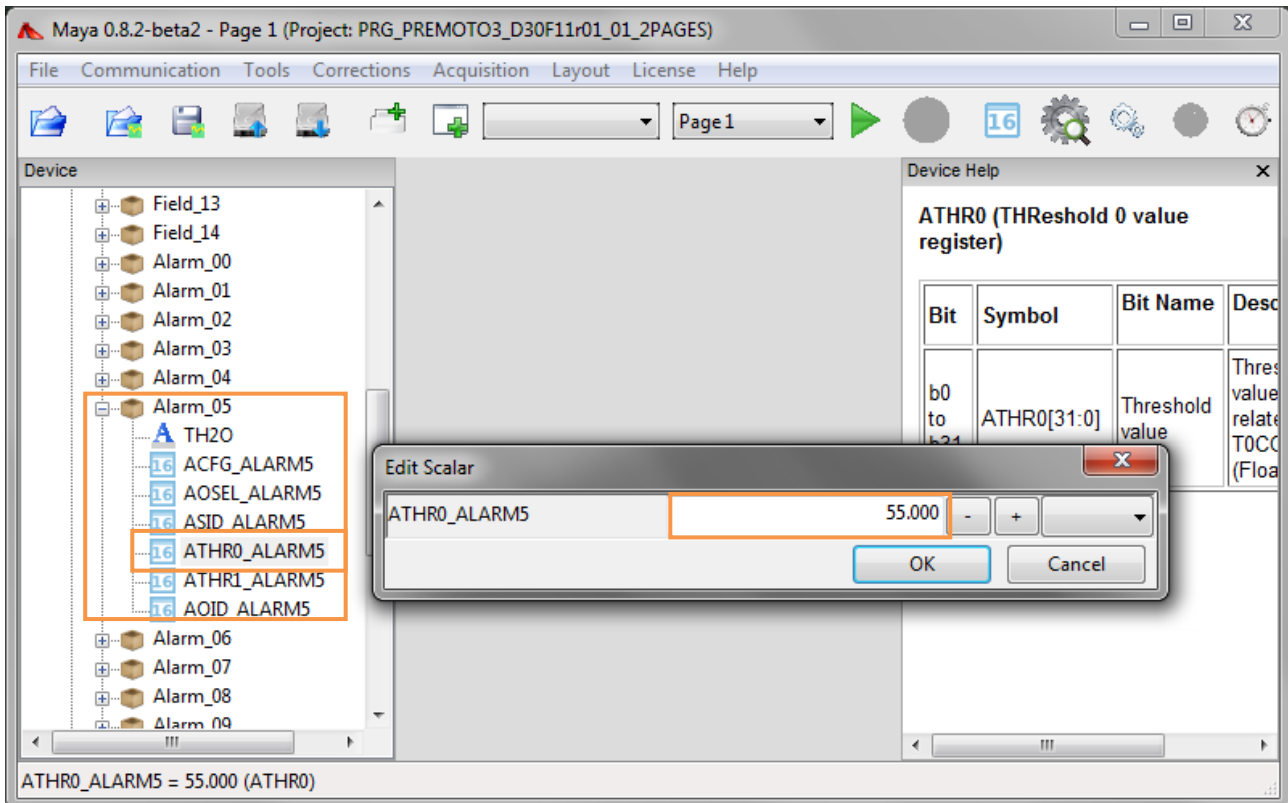
### NOTES:

- The values may be changed independently in any page
- The groups of alerts may vary according to the configuration of the dashboard

### 11.3 Changing the alarm thresholds

The procedure for changing the alarm thresholds is identical to the one seen for the changing flash. You should therefore insert the new values of the **ATHR0\_ALARM** and **ATHR1\_ALARM** scalars of the desired **Alarm** group.

The figure illustrates the alarm threshold (55°C) of the engine temperature channel (**TH2O**) configured on the **Alarm\_05** group



#### NOTES:

- The values may be changed independently in any page
- The groups of alerts may vary according to the configuration of the dashboard

## 11.4 Setting a CAN channel in reception

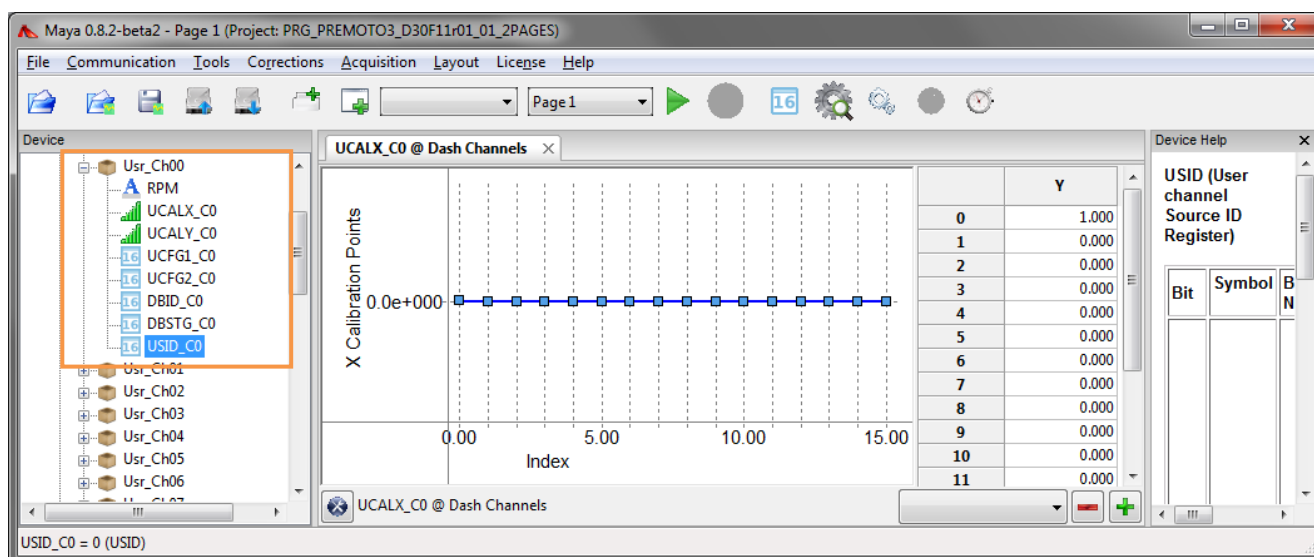
The example implies you want to configure a CAN channel (e.g. RPM) with the following features:

**CAN Bus Speed:** 1 Mb/s  
**CAN Byte Order:** Little Endian (Intel)  
**CAN ID:** 0x100  
**Start bit:** 0  
**Bit Length:** 16  
**Channel Type:** Unsigned  
**Calibration Type:** Linear (Gain+Offset)  
**Gain:** 1  
**Offset:** 0

You may also associate the channel to a group of the **Dash Channel Map** (e.g. **Usr\_Ch00**)

Set the scalars according to the following table:

D30EVO – Dash Channels Map			
GROUP	Name	Type	Description
Misc Register	CAN	Bitmask	Set the parameter <b>BAUD</b> at <b>1Mb/s</b> and the parameter <b>BO</b> as <b>Little Endian</b>
Usr_Ch00	Usr_Ch00	String	Set as <b>RPM</b>
	UCALX_C0	Vector	Set the first cell ( <b>Y0 = GAIN</b> ) at <b>1</b> and the second ( <b>Y1 = OFFSET</b> ) at <b>0</b>
	UCFG1_C0	Bitmask	Set the parameter <b>Calib</b> as <b>Line (gain+offset)</b> , the parameter <b>RXTYP</b> as <b>CAN</b> and the parameter <b>TXTYP</b> as <b>NULL</b> .
	UCFG2_C0	Bitmask	Set the parameter <b>SIGN</b> as <b>unsigned</b>
	DBID_C0	Value	Set at <b>100</b> (hexadecimal)
	DBSTG_C0	Bitmask	Set the parameter <b>SB</b> at <b>0</b> and the parameter <b>DL</b> at <b>16</b>
	USID_C0	Bitmask	Leave the parameter <b>ID</b> at <b>0</b> and the parameter <b>VAL</b> at <b>Normalized</b>



## 12 DEVICE TABLES

The following tables summarize the components of the **D30EVO Maya** device, divided into groups:

D30EVO – Dash Channels Map					
GROUP	Name	Type	Description	EVO lic.	ADV lic.
Misc Register	MCFG	Bitmask	Allows to setup the number of active pages ( <b>NPG</b> parameter), the start page ( <b>PA</b> parameter), the backlight ascent ( <b>BK</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CAN	Bitmask	Allows to setup the speed of the CAN bus ( <b>BAUD</b> parameter) and the byte transmission order ( <b>BO</b> parameter)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	SER	Bitmask	Allows to setup the speed of the serial bus ( <b>BAUD</b> parameter) and the byte transmission order ( <b>BO</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
From Usr_Ch00 to Usr_Ch31	Usr_Ch..	String	Name of the configured channel (not changeable by <b>EVO</b> license)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	UCALX_...	Vector	Channel calibration vector	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	UCALY_...	Vector	Channel calibration vector	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	UCGFG1_...	Bitmask	Allows to setup the channel calibration type ( <b>Calib</b> parameter), the channel source ( <b>RXTYP</b> parameter) and allows to send it to the desired bus ( <b>TXTP</b> parameter).	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	UCGFG2_...	Bitmask	Includes the required settings for the channel transmission, such as frequency ( <b>TXRATE</b> parameter) and sign parameter ( <b>SIGN</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	DBID	Value	Indicates the ID of receipt/transmission of the channel (in hexadecimal) – valid for channels on the CAN bus	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	DBSTG	Bitmask	Allows to setup the source byte of the channel ( <b>SB</b> parameter) and the length ( <b>DL</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	USID	Bitmask	Allows to setup the source channel (valid only if the channel is setup in transmission) and whether calibration is applied	<input type="checkbox"/>	<input checked="" type="checkbox"/>



D30EVO – Page1 ... Page5 Map					
GROUP	Name	Type	Description	EVO lic.	ADV lic.
Bar	BCFG	Bitmask	Allows to setup the multiplying factor of the engine route scale (standard values, multiple of 3)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	BLBL0 ... BLBL6	Value	Allows to setup the 7 values of the engine route scale, from the minimum value (BLBL0) to the maximum value (BLBL6)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	BSID	Bitmask	Allows to setup the number of the source channel of the route bar (ID parameter), view the normalised or raw value (VAL parameter) and the inclusion group (TYP parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_00	string	String	Label viewed in the <b>Field_00</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_01	string	String	Label of the channel viewed in the <b>Field_01</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CCFG	Bitmask	Allows to setup the channel type to view (number or string – <b>TYP</b> parameter), transfer the value ( <b>OFS</b> parameter), the name of decimal points ( <b>OFS</b> parameter) and whether to activate the view ( <b>FS</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CSID	Bitmask	Allows to setup the number of the channel to view ( <b>ID</b> parameter), whether to view the calibrated channel or as raw data ( <b>VAL</b> parameter) and to select origin ( <b>TYP</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_02	string	String	Label viewed in the <b>Field_02</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_03	string	String	Label of the channel viewed in the <b>Field_01</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CCFG	Bitmask	Allows to setup the channel type to view (number or string – <b>TYP</b> parameter), transfer the value ( <b>OFS</b> parameter), the name of decimal points ( <b>OFS</b> parameter) and whether to activate the view ( <b>FS</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CSID	Bitmask	Allows to setup the number of the channel to view ( <b>ID</b> parameter), whether to view the calibrated channel or as raw data ( <b>VAL</b> parameter) and to select origin ( <b>TYP</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_04	string	String	Label of the channel viewed in the <b>Field_01</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CCFG	Bitmask	Allows to setup the channel type to view (number or string – <b>TYP</b> parameter), transfer the value ( <b>OFS</b> parameter), the name of decimal points ( <b>OFS</b> parameter) and whether to activate the view ( <b>FS</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CSID	Bitmask	Allows to setup the number of the channel to view ( <b>ID</b> parameter), whether to view the calibrated channel or as raw data ( <b>VAL</b> parameter) and to select origin ( <b>TYP</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>

D30EVO – Page1 ... Page5 Map					
GROUP	Name	Type	Description	EVO lic.	ADV lic.
Field_05	string	String	Label of the channel viewed in the <b>Field_01</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CCFG	Bitmask	Allows to setup the channel type to view (number or string – <b>TYP</b> parameter), transfer the value ( <b>OFS</b> parameter), the name of decimal points ( <b>OFS</b> parameter) and whether to activate the view ( <b>FS</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CSID	Bitmask	Allows to setup the number of the channel to view ( <b>ID</b> parameter), whether to view the calibrated channel or as raw data ( <b>VAL</b> parameter) and to select origin ( <b>TYP</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_06	string	String	Label viewed in the <b>Field_06</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_08	string	String	Label viewed in the <b>Field_08</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_09	string	String	Label of the channel viewed in the <b>Field_01</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CCFG	Bitmask	Allows to setup the channel type to view (number or string – <b>TYP</b> parameter), transfer the value ( <b>OFS</b> parameter), the name of decimal points ( <b>OFS</b> parameter) and whether to activate the view ( <b>FS</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_10	string	String	Label viewed in the <b>Field_10</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_12	string	String	Label viewed in the <b>Field_12</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_13	string	String	Label of the channel viewed in the <b>Field_01</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CCFG	Bitmask	Allows to setup the channel type to view (number or string – <b>TYP</b> parameter), transfer the value ( <b>OFS</b> parameter), the name of decimal points ( <b>OFS</b> parameter) and whether to activate the view ( <b>FS</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	CSID	Bitmask	Allows to setup the number of the channel to view ( <b>ID</b> parameter), whether to view the calibrated channel or as raw data ( <b>VAL</b> parameter) and to select origin ( <b>TYP</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Field_14	string	String	Label viewed in the <b>Field_14</b> field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
da Alarm_00 a Alarm_15	string	String	Channel label – viewed in the alarm signalling field. Only changeable with <b>ADVANCE</b> license	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	ACFG_ALARM	Bitmask	Allows to define the condition of alarms ( <b>T0COND</b> , <b>T1COND</b> , <b>LOP</b> parameters), the view field ( <b>OUPT</b> parameter) and how to view alarms ( <b>OUT</b> parameter)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	AOSEL_ALARM	Bitmask	Allows to select the alarm led	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	ASID	Value	Allows to define the source channel of alarms	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	ATHR0_ALARM ATHR1_ALARM	Value	Allows to define the two alarm thresholds	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

D30EVO – Page1 ... Page5 Map					
GROUP	Name	Type	Description	EVO lic.	ADV lic.
LEDs	OCFG_BACK_LIGHT	Bitmask	Allows to select the behaviour ( <b>MD</b> parameter) and brightness ( <b>BRG</b> parameter) of the backlight	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	OCFG_LED1 ... OCFG_LED11	Bitmask	Allows to select the behaviour ( <b>MD</b> parameter) and brightness ( <b>BRG</b> parameter) of the associated led	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Digital_OUT	OCFG_DOUT0	Bitmask	Allows to select the behaviour ( <b>MD</b> and <b>BRG</b> parameters) of the DOUT0 digital output	<input type="checkbox"/>	<input checked="" type="checkbox"/>

[illegible]

NOTES:

[illegible]

NOTES:







# DATA ACQUISITION SYSTEMS

